

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

**Факультет інформатики та обчислювальної техніки
Кафедра обчислювальної техніки**

До захисту допущено:

Завідувач кафедри

_____ Сергій СТИРЕНКО

“ ____ ” _____ 2020 р.

Дипломний проект

на здобуття ступеня бакалавра

за освітньо-професійною програмою «Комп’ютерні системи та мережі»

спеціальності 123 «Комп’ютерна інженерія»

на тему: «Мобільний додаток під IOS платформу»

Виконав:

студент IV курсу, групи ІО-64

Єгор ОЛІЙНИЧЕНКО

Керівник:

Доцент, к.т.н.,

Віктор ПОРЄВ

Консультант з нормоконтролю:

Професор, д.т.н.,

Валерій СІМОНЕНКО

Рецензент:

Засвідчую, що у цьому дипломному проекті немає
запозичень з праць інших авторів без відповідних
посилань.

Студент _____

(підпис)

Київ – 2020 року

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

Факультет інформатики та обчислювальної техніки
Кафедра обчислювальної техніки

Рівень вищої освіти – перший (бакалаврський)

Спеціальність – 123 «Комп'ютерна інженерія»

Освітньо-професійна програма «Комп'ютерні системи та мережі»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____Сергій СТИРЕНКО

«___» _____ 2020 р.

ЗАВДАННЯ
на дипломний проект студента

Олійніченка Ігора Анатлоїдовича

1. Тема проекту «Мобільний додаток під IOS платформу»

керівник проекту Порєв Віктор Миколайович, к.т.н., доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від «07» травня 2020 року №1081-с

2. Термін здачі студентом закінченого проекту

3. Вихідні дані до проекту технічна документація

4. Зміст розрахунково-пояснювальної записки (перелік питань, які розробляються) _

Мобільний додаток під IOS платформу, визначення методології розробки мобільного додатку «Гаманець для криптовалюти», опис інструментальних засобів розробки.

5. Перелік графічного матеріалу

Принципова схема, функціональна схема та структурна схема _____

6. Консультанти проекту, з вказівкою розділів роботи, які до них вносяться

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
нормоконтроль	д.т.н., проф. Сімоненко В.П.		

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Найменування етапів дипломного проекту	Строк виконання етапів проекту	Примітки
1.	<i>Затвердження теми роботи</i>	<i>1.09.2019</i>	
2.	<i>Вивчення та аналіз завдання</i>	<i>2.09.2019-14.03.2020</i>	
3.	<i>Розробка архітектури та загальної структури програми</i>	<i>14.03.2020-25.03.2020</i>	
4.	<i>Розробка структур окремих Інтерфейсів програми</i>	<i>25.03.2020-2.04.2020</i>	
5.	<i>Програмна реалізація</i>	<i>2.04.2020-13.04.2020</i>	
6.	<i>Оформлення пояснювальної записки</i>	<i>13.04.2020-21.05.2020</i>	
7.	<i>Захист програмного продукту</i>	<i>21.05.2020 – 25.05.2020</i>	
8.	<i>Передзахист</i>	<i>26.05.2020</i>	
9.	<i>Захист</i>		

Студент

Єгор ОЛІЙНІЧЕНКО

Керівник

Віктор ПОРЄВ

ВІДОМІСТЬ ДИПЛОМНОГО ПРОЕКТУ

№ з/п	Формат	Позначення	Найменування	Кількість листів	Примітка
1.	A4		Завдання на дипломний проект	2	
2.	A4	ІАЛЦ.466500.002 ТЗ	Технічне завдання	4	
3.	A4	ІАЛЦ.466500.003 ПЗ	Пояснювальна записка	71	
4.	A4	ІАЛЦ.466500.004 А1	Принципова схема алгоритму	1	
5.	A4	ІАЛЦ.466500.005 А2	Функціональна схема	1	
6.	A4	ІАЛЦ.466500.006 А3	Структурна схема	1	

					ІАЛЦ.466500.001 ВП						
Зм.	Арк.	№ докум.	Підпис	Дата	Мобільний додаток під IOS платформу Відомість дипломного проекту			Літ.	Аркуш	Аркушів	
Розробив		Олійніченко Є.А.									
Перевірів		Порєв В.М.								1	1
Реценз.								НТУУ «КПІ», ФІОТ, ІО-64			
Н. Контр.		Сімоненко В.П.									
Затв.											

Технічне завдання до дипломного проекту

ЗМІСТ

1.	НАЙМЕНУВАННЯ ТА ОБЛАСТЬ ЗАСТОСУВАННЯ.....	2
2.	ПІДСТАВИ ДЛЯ РОЗРОБКИ.....	2
3.	МЕТА ТА ПРИЗНАЧЕННЯ РОЗРОБКИ.....	2
4.	ДЖЕРЕЛА РОЗРОБКИ.....	2
5.	ТЕХНІЧНІ ВИМОГИ.....	2
5.1.	Вимоги до програмної моделі мобільного додатку.....	2
5.2.	Вимоги до програмного забезпечення мобільного додатку.....	2

					ІАПЦ.466500.002 ТЗ				
Зм.	Арк.	№ докум.	Підпис	Дата	Мобільний додаток під IOS платформу Технічне завдання	Літ.	Аркуш	Аркушів	
Розробив		Олійніченко Є.А.							
Перевір.		Порєв В.М.					1	2	
						НТУУ “КПІ”, ФІОТ, ІО-64			
Н. контр.		Сімоненко В.П.							
Затверд.									

1. НАЙМЕНУВАННЯ ТА ОБЛАСТЬ ЗАСТОСУВАННЯ

Назва розробки: Гаманець для криптовалют

Область застосування: Мобільні платформи на базі IOS

2. ПІДСТАВИ ДЛЯ РОЗРОБКИ

Поширення популярності використання криптовалют у світі та обширне їх використання на ринку послуг.

3. МЕТА ТА ПРИЗНАЧЕННЯ РОЗРОБКИ

Зручність отримання доступу до перерахування, поповнення коштів та оплати послуг у будь-якій точці світу за допомогою телефона.

4. ДЖЕРЕЛА РОЗРОБКИ

Джерелом розробки є науково-технічна література, публікації в виданнях, довідники, публікації в Інтернеті

5. ТЕХНІЧНІ ВИМОГИ

5.1. Вимоги до програмної моделі мобільного додатку

Мобільний додаток має мати наступний функціонал:

- Перераховувати кошти на інші гаманці;
- Можливість поповнювати власний рахунок;
- Можливість спосереджувати власний баланс рахунку;

5.2. Вимоги до програмного забезпечення мобільного додатку

- Операційна система IOS 11 і новіше;
- Наявність IOS 11 і новіше. Сумісність з Iphone 6/6 plus і новіше, Ipad 4 mini і новіше, Ipod touch (6 покоління) та Ipod touch (7 покоління).

					ІАЛЦ.466500.002 ТЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		2

Анотація

В бакалаврській *дипломній* роботі було розкрито функціональні особливості мобільного додатку «Гаманець для криптовалюти» та реалізовані інструментальні засоби розробки, визначено методологію розробки мобільного додатку «Гаманець для криптовалюти».

Досліджено архітектуру мобільного додатку, наведено структуру та проведено тестування розробленого мобільного додатку у середовищі програмування Swift.

Abstract

At the bachelor's degree program there was demonstrated the functionality of a mobile device "Haman for cryptocryptovalut" and realized tools for its disassembling, the methodology of disassembling a mobile device "Haman for cryptocryptovalut".

The architecture of the mobile data base was reviewed, the structure and testing of the fragmented mobile data base at the medium program Swift was performed.

ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломного проекту

на тему: «Мобільний додаток під IOS платформу»

Київ – 2020

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ	2
ВСТУП	3
РОЗДІЛ 1 ТЕОРЕТИЧНІ АСПЕКТИ СТВОРЕННЯ МОБІЛЬНОГО ДОДАТКУ «ГАМАНЕЦЬ ДЛЯ КРИПТОВАЛЮТИ»	7
1.1 МОБІЛЬНИЙ ДОДАТОК: СУТНІСТЬ, ГЕНЕЗИС	7
1.2 БІТКОЇН, ЯК ОСНОВНА ЕЛЕКТРОННА ВАЛЮТА	13
Висновки до першого розділу	27
РОЗДІЛ 2 РОЗРОБКА МОБІЛЬНОГО ДОДАТКУ «ГАМАНЕЦЬ ДЛЯ КРИПТОВАЛЮТИ»	28
2.1 АНАЛІЗ І ХАРАКТЕРИСТИКА ОБ’ЄКТА ПРОЕКТУВАННЯ	28
2.2 ІНСТРУМЕНТАЛЬНІ ЗАСОБИ РОЗРОБКИ	30
2.3 АЛГОРИТМ РОЗРОБКИ МОБІЛЬНОГО ДОДАТКУ	34
2.4 АРХІТЕКТУРА МОБІЛЬНОГО «ГАМАНЕЦЬ ДЛЯ КРИПТОВАЛЮТИ».	35
Висновки до другого розділу	45
РОЗДІЛ 3 ПРОЕКТУВАННЯ МОБІЛЬНОГО ДОДАТКУ «ГАМАНЕЦЬ ДЛЯ КРИПТОВАЛЮТИ»	46
3.1 СТРУКТУРА МОБІЛЬНОГО ДОДАТКУ «ГАМАНЕЦЬ ДЛЯ КРИПТОВАЛЮТИ»	46
3.2 ІНСТРУКЦІЯ РОБОТИ КОРИСТУВАЧА З СИСТЕМОЮ	48
Висновки до третього розділу	54
ВИСНОВКИ	55
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	57

					ІАЛЦ.466500.003 ПЗ						
Зм.	Арк.	№ докум.	Підпис	Дата	Мобільний додаток під IOS платформу Пояснювальна записка			Літ.	Аркуш	Аркушів	
Розробив		Олійніченко Є.А.								1	71
Перевір.		Порєв В.М.									
Н. контр.		Сімоненко В.П.									
Затверд.					НТУУ “КПІ” ФІОТ, ІО-64						

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

БД	база даних
ІС	інформаційні системи
ОС	операційна система
ПЗ	програмне забезпечення
ПК	персональний комп'ютер
USIM	модуль ідентифікації абонента мережі UMTS
EDGE	Enhanced Data Rates for GSM Evolution
HSDPA	High- Speed Downlink Packet Access
IMT- 2000	International Mobile Telecommunications 2000

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		2

ВСТУП

Актуальність дослідження. Незважаючи на величезний прогрес в економічній глобалізації, світова фінансова система все ще дуже фрагментована, і лише деякі її гравці отримують реальний шанс на щось вплинути. Очевидно, що блокчейн має відмінний потенціал для консолідації і, можливо, навіть стандартизації фінансових ринків по всьому світу.

Різні приватні підприємства і регулюючі органи вже почали інтегрувати технологію блокчейна в свої бізнес-моделі.

Одна з нагальних проблем світової економіки – уповільнення економічного зростання. Хоча більшість світових проблем, про які ми чуємо щодня, дуже серйозні, недолік економічного зростання – одна з проблем, яка може посилити інші труднощі, з якими стикається світ.

Багато світових організацій визнають проблему зниження економічного зростання. Доповідь Організації Об'єднаних Націй починається з заголовка – «Глобальна економіка залишається втягнутою в тривалий період уповільненого зростання». Далі в доповіді підкреслюється:

Дослідження, проведене Світовим банком, стверджує наступне – «в останні роки довгострокове фінансування привертає підвищену увагу законодавців, дослідників та інших зацікавлених сторін фінансового сектора». Відсутність довгострокового фінансування «ще більше посилює положення існуючих вразливостей і збільшило вплив довгострокових фінансових проблем, зокрема – для інфраструктури».

У дослідженні, проведеного Організацією економічного співробітництва та розвитку (ОЕСР), також відзначається, що «довгострокові інвестиції відіграють ключову роль у сприянні зростанню та створенні робочих місць».

Також у дослідженні зазначається, що «хоча довгострокові інвестиції необхідні для росту, їх фінансування стикається з серйозними проблемами, які вимагають більшій участі корпоративних інвесторів» – приватні підприємства повинні більш

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		3

активно брати участь у вирішенні проблем, пов'язаних з дефіцитом довгострокового фінансування.

А соціальна напруженість, як правило, швидко переростає у всілякі конфлікти, що посилює проблему ще сильніше.

G30, консультативна група з міжнародних економічних та монетарних питань, провела дуже ретельне дослідження, щоб визначити чотири основні принципи, що впливають на надання довгострокових фінансів.

До групи G30 входять провідні світові лідери з приватного і державного секторів, а також такі вчені, як Пол А. Волкер (Paul A. Volcker), Жан-Клод Тріше (Jean-Claude Trichet), Бен Бернанке (Ben Bernanke), Маріо Драгі (Mario Draghi), Тімоті Гейтнер (Timothy Geithner) і Пол Кругман (Paul Krugman). Ось чотири висновки, до яких прийшла ця група:

Фінансова система повинна спрямовувати заощадження домашніх господарств і корпорацій в адекватну можливість фінансування з тривалим терміном його повернення, щоб задовольнити зростаючі інвестиційні потреби реальної економіки.

Довгострокове фінансування має здійснюватися суб'єктами з довгостроковими перспективами.

Для підтримки довгострокових інвестицій повинен бути доступний широкий спектр фінансових інструментів.

Ефективна світова фінансова система повинна забезпечувати економічне зростання за допомогою стабільних міжнародних потоків довгострокового фінансування, підтримуваних належним глобальним регулюванням.

Технологія блокчейна і її агресивна популяризація за допомогою швидкого зростання цін на криптовалюту може допомогти світовій фінансовій системі розв'язати цю проблему головної проблеми, яка полягає в дефіциті довгострокового фінансування і забезпечити подальше глобальне зростання економіки за рахунок того, що технологія блокчейна знаходиться в повній відповідності з кожним з чотирьох принципів, сформульованих G30.

Мета та завдання дослідження. Метою даної дипломної роботи є створення мобільного додатку «Гаманець для криптовалюти».

Завданнями виступають:

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		4

- розкриття функціональних особливостей мобільного додатку «Гаманець для криптовалюти»;
- опис інструментальних засобів розробки;
- визначення методології розробки мобільного додатку «Гаманець для криптовалюти»;
- дослідження архітектури мобільного додатку «Гаманець для криптовалюти»;
- наведення структури мобільного додатку «Гаманець для криптовалюти»;
- тестування розробленого мобільного додатку «Гаманець для криптовалюти»;
- верифікація результатів дослідження.

Об’єкт та предмет дослідження. Об’єктом роботи виступає мобільний додаток «Гаманець для криптовалюти».

Предметом є процес розробки мобільного додатку «Гаманець для криптовалюти».

Методи дослідження. Для вирішення поставлених завдань використовувалися як загальнонаукові, так і спеціальні методи наукового пізнання. Системний аналіз, дедуктивний та індуктивний методи використовувалися при визначенні суті понять «мобільний клієнт серверний додаток», «фінансовий актив», «блокчейн».

Наукова новизна отриманих результатів.

- запропоновано комплексне рішення реалізації мобільного додатку «Гаманець для криптовалюти»;
- наведено структуру та архітектуру мобільного додатку «Гаманець для криптовалюти».

Практичне значення отриманих результатів. Результати роботи можуть бути використані при розробці, встановленні та налаштуванні мобільного додатку «Гаманець для криптовалюти».

Апробація отриманих результатів...

Публікації...

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		5

Достовірність дослідження. При реалізації проекту було вивчено значне число літературних джерел, посібників і статей, що охоплюють обрану тематику. Повний їх список представлений в кінці проекту.

Структура дипломної роботи. Робота складається зі вступу, трьох розділів, що включають в себе 11 підрозділів, висновків та списку літератури з 46 найменувань, загальний обсяг роботи 98 сторінок.

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		6

РОЗДІЛ 1

ТЕОРЕТИЧНІ АСПЕКТИ СТВОРЕННЯ МОБІЛЬНОГО ДОДАТКУ «ГАМАНЕЦЬ ДЛЯ КРИПТОВАЛЮТИ»

1.1 Мобільний додаток: сутність, генезис

У сучасному світі стільниковий зв'язок настільки проник в життя людини, що більшість вже не уявляють своє життя без телефону. Ще вчора найпопулярнішими функціями "мобільника" були дзвінки і відправлення SMS. Але сьогодні пріоритети починають зміщуватися кілька в іншу сторону. Сучасний телефон – це вже не просто засіб зв'язку людини з людиною, це багатофункціональний пристрій, який допомагає не тільки спілкуватися, але й вчитися, заробляти, пізнавати світ, розважатися. Безумовно, це стало можливим завдяки розвитку мобільного інтернету і безпосередньо мобільних додатків [8].

Мобільним додатком можна назвати комп'ютерну програму, створену спеціально для використання в мобільному телефоні, смартфоні або комунікаторі, яка призначена для виконання того чи іншого завдання.

Історія мобільних додатків налічує вже більше десяти років. За великим рахунком відправною точкою для створення мобільних додатків стала поява на мобільному телефоні екрану. Природно, перше програмне забезпечення для телефонів представляло собою вбудовані додатки, які призначалися для виконання конкретних функцій телефону і встановлювалися в пристрій самими виробниками.

Мабуть, першим мобільним додатком, крім ПЗ, що відповідає безпосередньо за роботу телефону, стала телефонна книжка – та частина програмного забезпечення апарату, яка впорядковувала контакти користувача [2]. Спочатку в записну книжку можна було занести лише ім'я та номер телефону абонента. Але поступово до цього додатка додавалися нові функції – крім імені та телефону, стало можливим занесення адреси, електронної пошти та інших даних того чи іншого абонента.

З появою можливості обміну короткими текстовими повідомленнями (SMS) в телефон додався ще один додаток, що дозволяє писати, редагувати, відправляти невеликі електронні тексти.

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		7

Час появи першого мобільного додатка, встановленого на телефон поверх вже наявного програмного забезпечення, можна віднести до кінця 90-х років минулого століття, коли стільниковий зв'язок став поступово входити в життя мільйонів людей в усьому світі. Варто зазначити, що на той час, виробники телефонів вже чітко уявляли, що софт для "мобільника" – це перспективний напрям, як з точки зору розробки технологій, так і з точки зору їх окремого комерційного використання. Тоді в програмну оболонку стільникових телефонів, крім самих необхідних додатків, виробники стали встановлювати додаткове ПЗ. Як правило, це були різні мультимедійні додатки – невеликі аркадні ігри, редактори рингтонів, калькулятори, календарі тощо [30]

З появою в 1997 році на ринку стільникового зв'язку технології WAP, що дозволяє виходити в інтернет за допомогою мобільного телефону, кількість програмних додатків, як і їх розробників, стало зростати. Справа в тому, що тепер поставити на телефон будь-яку програму стало набагато зручніше і простіше, так як раніше установка додатку могла бути здійсненна лише за допомогою DATA-кабелю, що з'єднує стаціонарний комп'ютер або ноутбук з мобільним телефоном. Варто відзначити, що в той час не кожна модель стільникового телефону поставлялася в комплекті з DATA-кабелем, що обмежувало використання мобільних додатків.

Можливість виходу в інтернет безпосередньо через телефон дозволила встановлювати на апарат різне ПЗ, а також ігри навіть тим людям, які не мали домашнього комп'ютера. Крім того, WAP міг функціонувати навіть на бюджетних телефонах, за рахунок чого число користувачів мобільних додатків також збільшувалося. Єдиним мінусом WAP-доступу стала висока вартість передачі даних – скачавши тільки кілька програм для телефону, користувач міг витратити весь свій рахунок.

До початку нового тисячоліття почався бурхливий розвиток ринку мобільного контенту в цілому і мобільних додатків зокрема. Поява нових технологій передачі даних за допомогою стільникового зв'язку (GPRS, EDGE) дозволяє здешевити мобільний інтернет-трафік. На той час ринок мобільних пристроїв стільникового зв'язку стали поступово завойовувати смартфони і комунікатори. Володіючи більш широкими можливостями і продуктивністю, вони відрізнялися від звичайних

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		8

мобільних телефонів наявністю досить розвиненої операційної системи (Windows Mobile, Simbian, RIM, Android, Mac OS), яка є відкритою для розробки програмного забезпечення сторонніми розробниками, на відміну від програмного середовища звичайних мобільних телефонів, яка закрита для сторонніх розробників. При цьому варто відзначити, що установка додаткових додатків дозволяє значно поліпшити функціональність смартфонів і комунікаторів у порівнянні зі звичайними мобільними телефонами. Але, зауважимо, що кожна конкретна операційна система вимагає установки відповідних саме їй додатків, тобто створених спеціально для тієї чи іншої ОС програм. Наприклад, програму-органайзер створену для Windows Mobile можна встановити на ОС Simbian, яка застосовується в смартфонах Nokia, або Mac OS (Mobile Touch version) для телефону iPhone від компанії Apple [9].

До того ж, варто відзначити, що наявність повнофункціональної операційної системи не робить смартфони і комунікатори більш привабливими в очах більшості користувачів. Сучасні стільникові телефони, а точніше моделі, що відносяться до середньої цінової категорії і вище, часто цілком можуть впоратися з багатьма завданнями. Вони можуть працювати з електронною поштою, переглядати текстові документи і електронні таблиці, фотографії та відеофайли.

Необхідно також розуміти, що програми, написані спеціально для операційної системи смартфона або комунікатора, є повноцінними скомпільованими в двійковий код послідовностями низькорівневих мікропроцесорних команд. А в силу того, що у всіх смартфонах і комунікаторах встановлені більш потужні процесори, ніж в мобільних телефонах, можливості програміста по створенню таких додатків практично не обмежені.

Сьогодні, крім незалежних інтернет-майданчиків для поширення мобільних додатків, власні подібні ресурси створюють провідні виробники стільникових телефонів, смартфонів і комунікаторів. Деякі з них вже мають або планують в найближчому майбутньому відкрити власний майданчик з продажу мобільних додатків. Такі спеціалізовані портали з розповсюдження мобільного ПЗ вже мають виробник телефонів iPhone – компанія Apple (App Store), компанія Nokia (OVi), виробник смартфонів Blackberry – компанія RIM (App World і Application Center), компанія Google (Android Market), компанія Sony– Ericson (PlayNow arena) і ряд

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		9

інших. Крім самих додатків, ці онлайн-ресурси продають також різноманітний мобільний контент (музику, відео, картинки, електронні книги і т.д.).

Крім того, виробники мобільних телефонів, смартфонів і комунікаторів і творці операційних систем, а також оператори стільникового зв'язку випускають програмні інструменти для створення мобільних додатків. Вони пропонуються на основі як відкритого, так і закритого ПЗ і часто призначені для створення програмного забезпечення для конкретної операційної системи, що часто означає – для конкретної моделі мобільного пристрою (наприклад, iPhone, гуглфон або смартфони Nokia).

«Клієнт – сервер» (англ. Client-server) – обчислювальна або мережева архітектура, в якій завдання або мережеве навантаження розподілені між постачальниками послуг, званими серверами, і замовниками послуг, званими клієнтами. Фактично клієнт і сервер – це програмне забезпечення.

Архітектура програми часто створюється для ілюстрації загальних властивостей програмного забезпечення (наприклад, код програми та платформа) і обладнання (наприклад, клієнт, сервер і мережеві пристрої). У літературі виділяють ряд базових шаблонів.

Архітектура додатків, як правило, проектується в термінах архітектури «клієнт-сервер», при якій один або кілька клієнтських пристроїв подають запит на інформацію з сервера. Сервер зазвичай володіє необхідною інформацією (рис. 1.1).

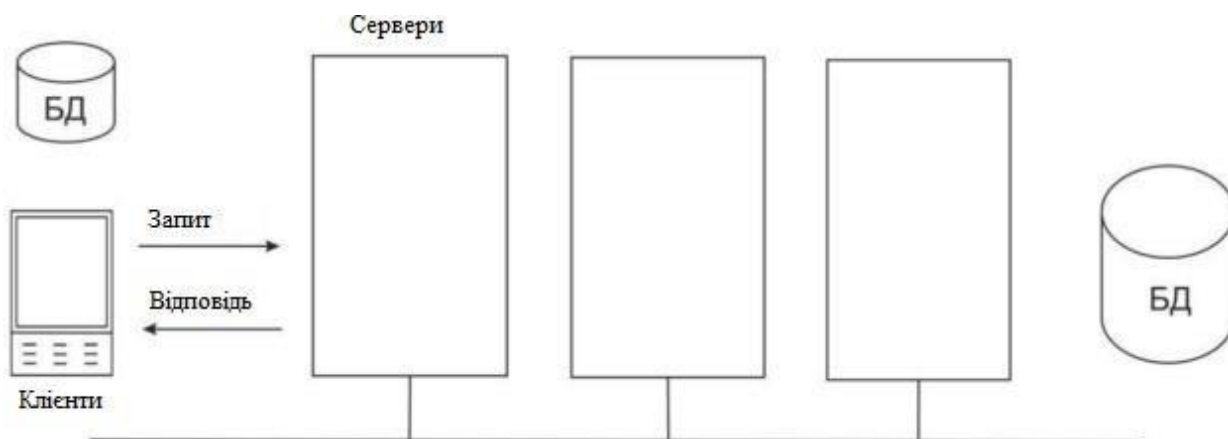


Рисунок 1.1 – Клієнт-серверна архітектура [26]



Рисунок 1.2 – Дволанкова архітектура



Рисунок 1.3 – Триланкова архітектура

Розглянемо такі поняття клієнт-серверної архітектури: шари, рівні і зв'язок між шарами і рівнями.

Функціональність коду по всьому додатку необов'язково рівномірна. Деякі розділи коду програми краще підходять для обробки користувальницького інтерфейсу, в той час як інші розділи розроблені для управління бізнес-логікою або з'єднання з базою даних або серверних систем.

Розшарування описує поділ робіт всередині коду програми на одній машині. Найчастіше шари – це не більше ніж програмні модулі, які розміщені в різних папках або каталогах на стороні клієнта або сервера.

З боку клієнта зазвичай є від нуля до трьох шарів в коді програми. З боку сервера – від одного до трьох шарів коду програми. Частково це важливо для гарного проектування програмного забезпечення, яке гарантує повторне використання коду, частково це важливо для безпеки, частково з міркувань зручності.

Клієнт з нульовою кількістю шарів коду, по суті, не має спеціального коду програми. Цей тип клієнта зазвичай згадується як тонкий клієнт і можливий в архітектурі клієнт-сервер, при якому сервер містить весь призначений для користувача код програми. Клієнт з одним до трьох шарів коду програми зазвичай називається товстим клієнтом.

Сервер, також може містити від одного до трьох шарів спеціального коду програми. Проте, за визначенням, не може бути нульової кількості шарів коду на сервері.

Шар, код якого найтісніше взаємодіє з користувачем, часто згадується в літературі як шар уявлення. Другий шар зазвичай обробляє бізнес-логіку коду. Третій шар називають рівнем доступу до даних (зв'язок з базою даних або з джерелом даних).

Цілком можлива наявність більш ніж трьох рівнів на стороні клієнта або сервера, але занадто багато шарів може привести до труднощів при управлінні.

1.2 Біткоїн, як основна електронна валюта

Криптовалюта – один з видів цифрової валюти. Її емісія і облік засновані на різних криптографічних методах, її функціонування відбувається децентралізовано, в розподіленій комп'ютерній мережі. Найпопулярніша зараз криптовалюта – біткоїни. Друга за популярністю криптовалюта – лайткоїн. Ця система виникла як відгалуження біткоїнів в жовтні 2011 року, її засновником виступив колишній співробітник компанії Google Чарлі Лі. На сайті валюти зазначається, що вона відрізняється більш швидким часом підтвердження транзакцій і підвищеною ефективністю зберігання в порівнянні з біткоїнами. Також там говориться, що «з серйозною галузевою підтримкою, торговим оборотом і ліквідністю Litecoin є перевіреним засобом для комерції на додаток до біткоїнів». Загальна капіталізація Litecoin – понад 2,2 млрд дол.

Біткоїни і лайткоїн можна порівняти з золотом і сріблом. На сьогоднішній день налічується вже з десятків більш-менш великих криптовалют, а в цілому близько 700. Але саме біткоїни розвиваються найбільш інтенсивно.

Третя криптовалюта на сьогоднішній день – Ether, показник капіталізації для неї становить близько 21 млрд дол. Цю криптовалюту часто називають «ефір». Криптовалюта була створена як спосіб краудфандінгового фінансування проекту Ethereum канадського програміста російського походження В. Бутеріна. Ethereum замислювався як відкрита платформа з вбудованою мовою програмування, основна ідея якої в тому, щоб будь-який програміст міг створювати додатки на базі технології блокчейн. На момент випуску в 2015 році ефір коштував 2,9 дол., нині він оцінюється більш ніж в 224 дол. на четвертому місці за обсягом ринкової капіталізації знаходиться Ріпл – Ripple. Протокол Ripple забезпечує миттєвий і прямий переказ коштів між двома сторонами в будь-якому вигляді, а комісія за нього встановлена на мінімальному рівні. Система була запущена в 2012 році і привернула серйозну увагу банків. За даними на квітень 2017 року, серед клієнтів Ripple - 75 банків, в тому числі такі гіганти, як BBVA, Standard Chartered і Bank of America. Капіталізація Ripple становить близько 6,7 млрд дол. [1]

Курс біткоїни швидко зростає. Відомо, що один з жителів США в 2010 році купив піцу за 10 000 біткоїни, що зараз еквівалентно понад 5 млн доларів. На

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		13

сьогоднішній день ринок даної віртуальної валюти налічує 15 404 700 одиниць в обігу, а обмінний курс біткоїни з моменту створення однойменної пирингової платіжної системи на початок 2016 р досяг 416,28 дол. за «монету», в серпні 2017 року – оновив історичний максимум, досягнувши позначки в 4367,71 дол. за віртуальну монету. Капіталізація валюти перевищила 72 млрд дол. Вартісна оцінка біткойнів істотно зросла в порівнянні з іншими світовими валютами. Очікується, що біткойн може стати новим золотим стандартом, який забезпечить грошову рівновагу в наявній фінансовій системі і представить нову міру вартості.

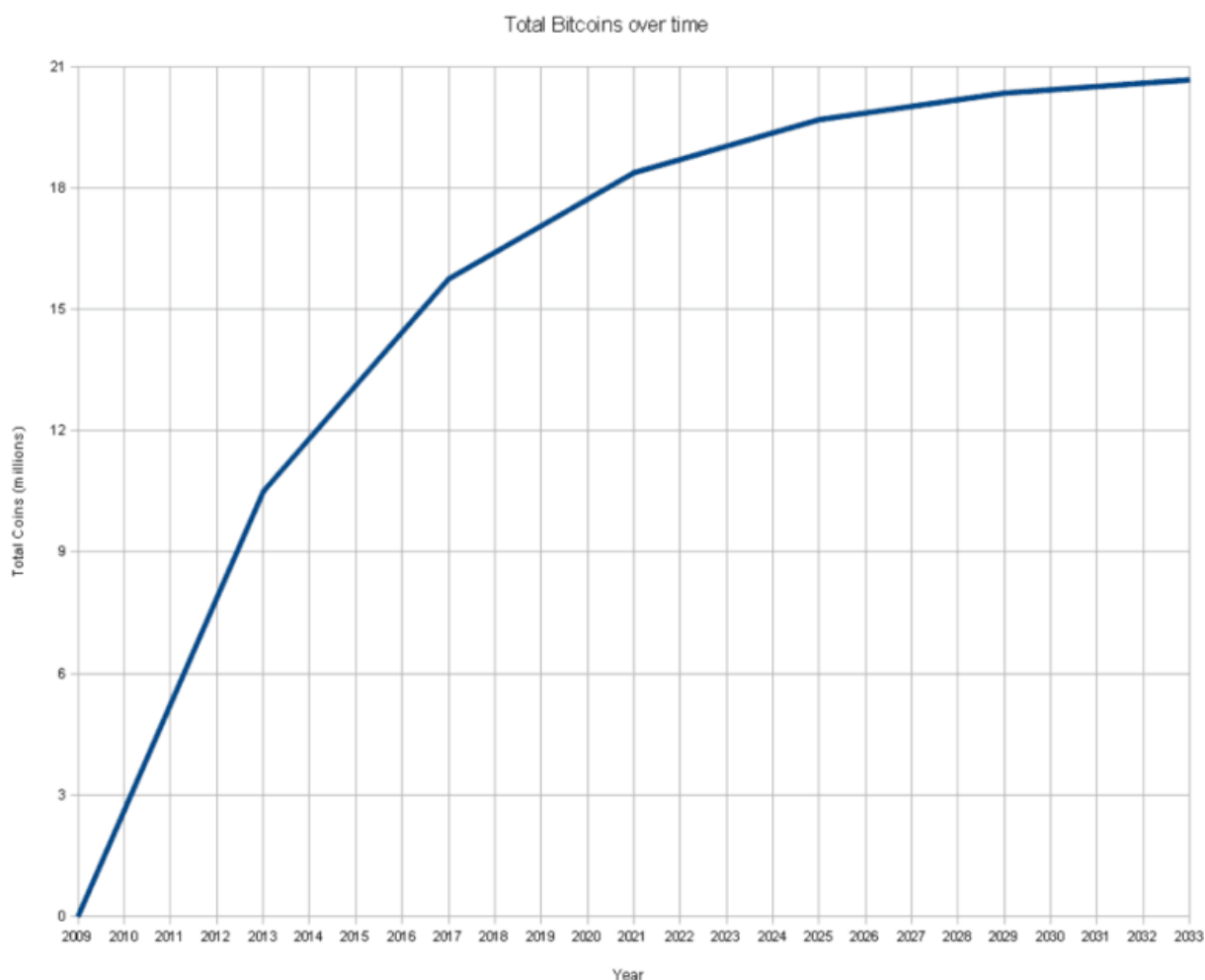


Рисунок 1.4 – Кількість біткойнів за 2009-2033 роки

У серпні 2017 року було прийнято рішення про розділення криптовалюти біткоїни на дві нові валюти: власне біткоїни і біткоїни кеш. Це сталося в результаті відмови частини користувачів біткойнів від переходу на новий протокол Segwit2x, який обґрунтовували необхідністю зниження комісії за транзакції і низькою

швидкістю проведення платежів. Щоб встановити новий протокол SegWit2x, необхідно було оновити програмне забезпечення, яке використовується Майнер і власниками біткоїни. Після цього розмір блоку біткоїни (запис про минулу транзакцію) мав би збільшитися з 1 Мб до 2 Мб. Оновлення було анонсовано в середині липня 2017 р. проте частина користувачів заявила, що не має наміру його встановлювати. Так відбулося розділення криптовалюти.



Рисунок 1.5 – Лінійний графік курсу Bitcoin по відношенню до USD

Швидке зростання виробництва і застосування нової криптовалюти привертає увагу. У багатьох фахівців виникає питання про вплив цієї системи як на світову фінансову систему, так і на фінанси окремих країн.

Confirmed Transactions Per Day

The total number of confirmed transactions per day.

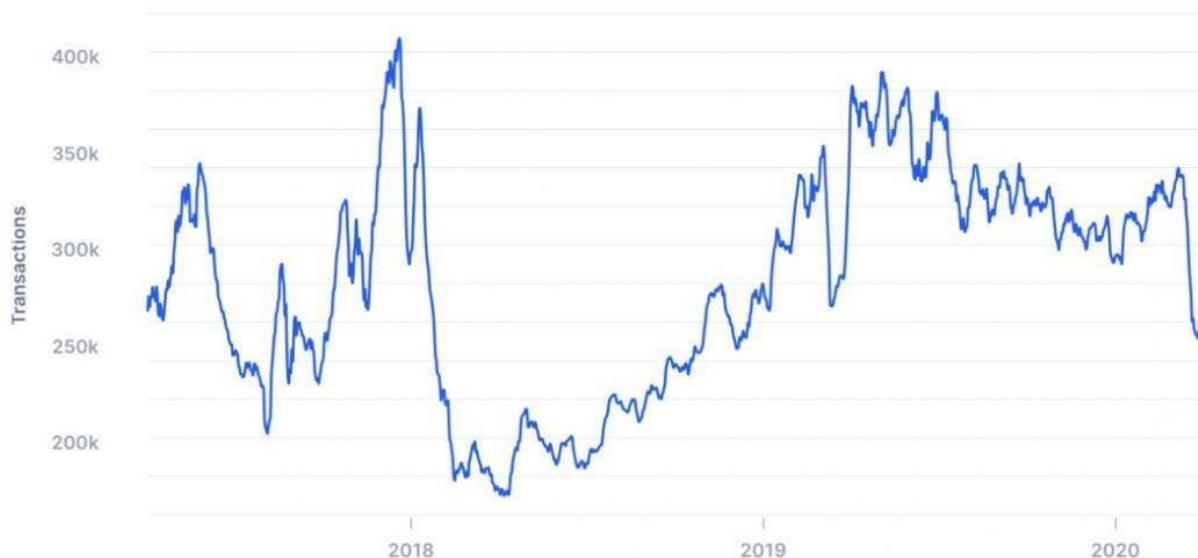


Рисунок 1.6 – Кількість транзакцій біткоїнів в місяць

У світі відсутній єдиний підхід до оцінки цього впливу. В одних країнах оборот біткоїнів заборонений законодавчо, як наприклад в Росії, в інших вже встановлюють вуличні банкомати з обміну цифрової валюти на гроші. Криптовалюту біткоїни не можна назвати фінансовою пірамідою. Криптовалюта – це реальний програмний продукт, зростання курсу якого залежить тільки від попиту і пропозиції, а не від подальших вкладників.

Біткоїни (bitcoin, btc, БТК, біткоїн) – нове покоління децентралізованої цифрової валюти, створеної і діючої тільки в мережі Інтернет. Найбільш важлива відмінність біткоїнів від всіх інших форм грошей – децентралізація. Жодна установа в світі не контролює біткоїни. Жоден банк не може контролювати емісію криптовалюти, яка відбувається за допомогою роботи мільйонів комп'ютерів по всьому світу, використовуючи програму для обчислення математичних алгоритмів. Саме в цьому полягає суть біткоїнів [11].

Ця криптовалюта ґрунтується на двох якостях – абсолютної анонімності і надійності внутрішньосистемних транзакцій. Біткоїни не пов'язані з банками, державами, запасами золота, так як є лише набором символів, які є результатом вирішення завдання. За задумом творця, на його курс може впливати тільки попит, який свідомо обмежений.

Специфіка нової валюти в тому, що вона не друкується центральним банком і не працює за його правилами. Банки можуть надрукувати скільки завгодно грошей, щоб покрити державний борг, тим самим знецінюючи свою валюту. А біткоїни вільні від цієї небезпеки. Емісія біткоїнів можлива тільки в цифровому вигляді і будь-який бажаючий може почати добувати або як кажуть «Майн» біткоїни в будь-який час. Майнінг біткоїни відбувається за допомогою використання обчислювальних потужностей комп'ютера в розподіленій мережі.

Але «добути» мільярд біткоїнів не можна, оскільки в коді біткоїнів є обмеження, «добути» максимум можна 21 мільйон біткоїнів. Незважаючи на це, біткоїн може ділитися до нескінченності на більш дрібні частини, до 1 Сатоши це 0,00000001 btc (ім'я «Сатоши» дано на честь творця).

Біткоїни приймають в якості оплати вже сотні сервісів і магазинів. Можна зробити висновок, що так само, як свого часу електронна пошта обмежила до мінімуму функціонування традиційної, а Skype потіснив телефонні компанії, біткоїни зможуть зайняти своє місце в платіжних системах.

Важливо відзначити універсальність нової валюти. Якщо в реальному житті мандрівникам доводиться користуватися тією чи іншою валютою, в залежності від країни проживання або перебування, то дана криптовалюта не належить жодній країні [3].

Віртуальні гроші біткоїни – це нова криптовалюта, яку створив нікому не відомий програміст (або група програмістів) під псевдонімом Сатосі Накамото. Сталося це в 2008 році. Термін «біткоїни» і алгоритм роботи придумав саме він. Ніхто не знає, звідки ця людина і як його звуть в реальному світі. Він створив не тільки програму, а й спеціальний додаток – електронний гаманець на персональному комп'ютері, що містить криптовалюту біткоїни [18]. Нові криптомонети – біткоїни або будь-яка інша криптовалюта створюються за допомогою процесу, званого Майнінг. Це єдина техніка, що дозволяє стати володарем електронного платіжного інструменту, пов'язана з рішенням комп'ютерних математичних задач. На створення криптовалюти йде величезна кількість електрики. Її досить було б для забезпечення 31 тисячі приватних будинків.

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		17

Суть виробництва біткоїнів зводиться до вирішення якоїсь складної кріптозадачі, яка вирішується спеціальним методом (методом повного перебору). Але звичайний комп'ютер для виконання цього завдання не підійде. Зазвичай Майнер криптовалюти використовують надпродуктивні комп'ютери або потужні сервери, що застосовуються для вирішення складних математичних головоломок. В даний час, переможець нагороджується 25 біткоїнами і визначається раз в 10 хвилин. Але оскільки мережа біткоїнів росте небувалими темпами, видобуток став технологічно складним процесом.

Криптовалюта біткоїни володіє наступними важливими функціями: засіб обміну; засіб заощадження; розрахункова одиниця.

Криптовалюту біткоїни можна використовувати для покупки товарів і послуг в інтернеті анонімно. Крім того, можна здійснювати міжнародні платежі відносно легко і дешево, тому що біткоїни не прив'язані до певної країни. Для зберігання біткоїни зазвичай використовується електронний гаманець. Він встановлюється і створюється на персональному комп'ютері. Зазвичай його шифрують, щоб уникнути злому. Однак тут є свої мінуси, якщо користувач забуде пароль для входу в гаманець або на комп'ютері вийде з ладу жорсткий диск, то назавжди може загубитися доступ до засобів. Але електронний гаманець має свої переваги над оффлайн версією. Користувач може отримати доступ до нього, скориставшись не тільки персональним комп'ютером, але і планшетом або телефоном. Специфіка схожа зі звичайними Ківі-гаманцями, Вебмані або інтернет-банкінгом. Одна з основних проблем цих гаманців в тому, що всі дані зберігаються на сервері. Якщо сервер зламають, то зламають і всю інформацію [19].

Біткоїни можуть бути передані будь-кому, хто повідомить коректну біткоїн-адресу або відкритий ключ. Для передачі біткоїнів поточний власник створює нову транзакцію, яка, крім вказівок про кількість переданих біткоїнів, містить підписаний ініціатором хеш попередньої транзакції, по якій біткоїни були отримані. Попередня транзакція стає «входом» поточної транзакції. Також вказується публічний ключ або біткоїн-адреса нового одержувача («вихід») (рис. 1.7). Транзакція широкомовним запитом по відкритих каналах без шифрування відправляється в мережу. Решта вузлів мережі, перш ніж прийняти транзакцію до обробки, перевіряють підписи.

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		18

Правильність підпису свідчить, що ініціатор дійсно є власником секретного ключа для адреси «виходу».

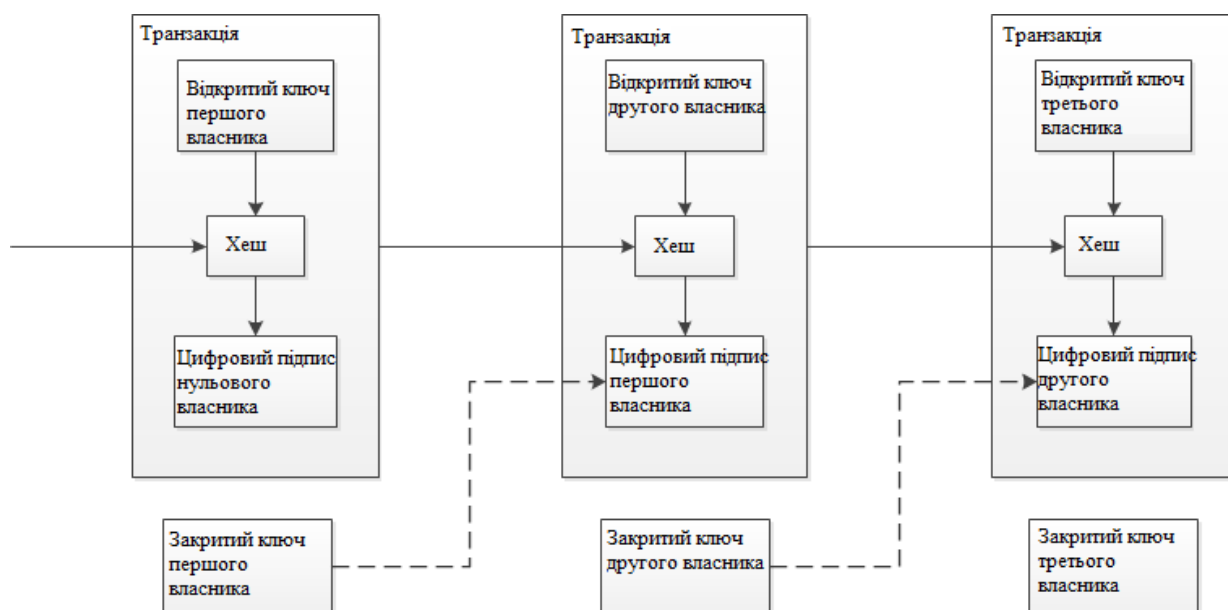


Рисунок 1.7 – Спрощена структура послідовних транзакцій з одним входом і одним виходом [33]

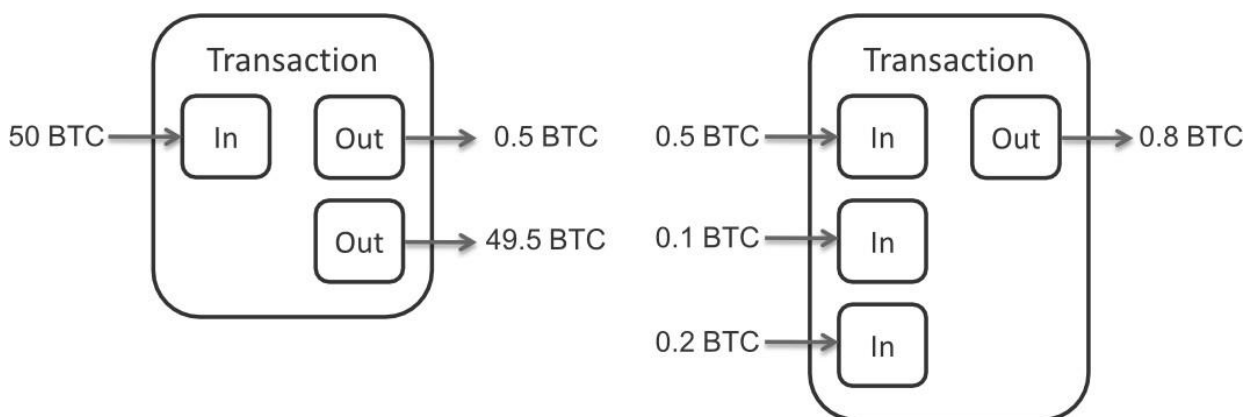


Рисунок 1.8 – Приклади множинних входів і виходів у транзакціях

Окремі транзакції об'єднують разом з іншими транзакціями в спеціальну структуру – блок. Інформація в блоках відкрита, не шифрується, її можна швидко перевірити ще раз.

Кожен блок завжди містить свій порядковий номер і хеш попереднього блоку. Всі блоки можна вибудувати в один ланцюжок, який містить інформацію про всі

скоєні операції з біткоїнів. З ними можна ознайомитися, наприклад, на спеціалізованих сайтах – браузерях ланцюжків блоків [8] (англ. Blockchainexplorer).

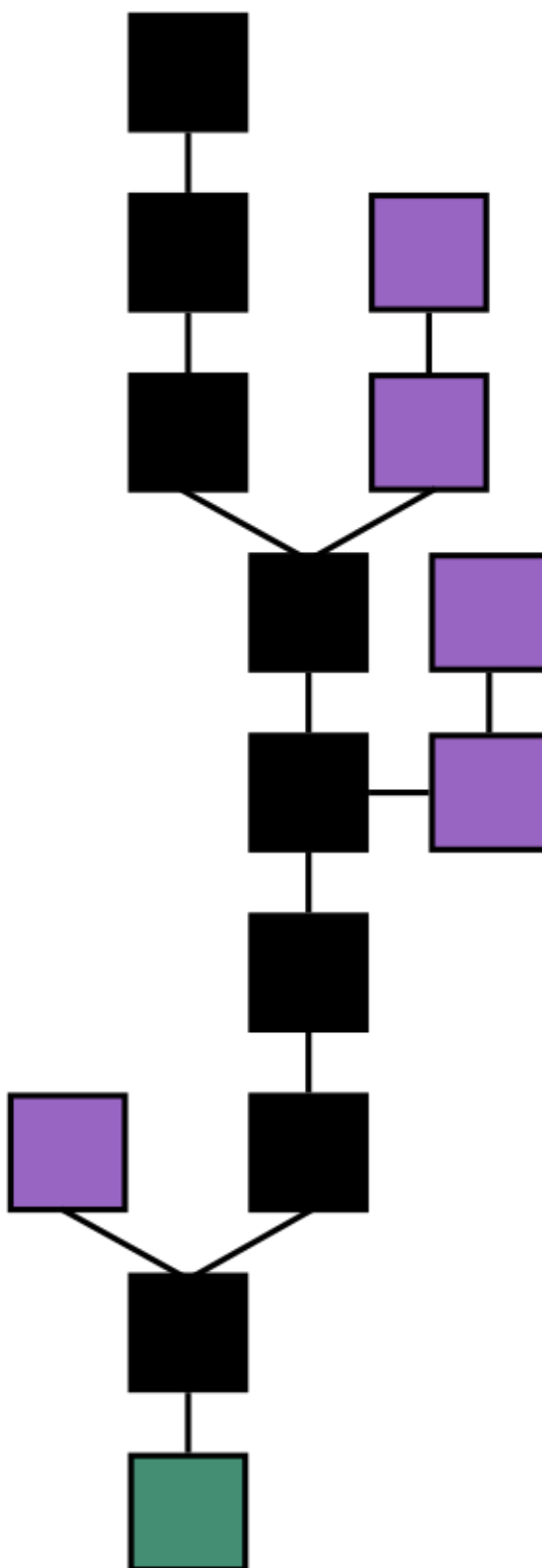


Рисунок 1.9 – Основна послідовність блоків (чорні) є найдовшою від початкового (зелений) до поточного. Побічні гілки (фіолетові) відсікаються

Перша транзакція в блоці завжди формується автоматично і передає винагороду за створення блоку [16]. Решта наповнення блоку беруть з черги транзакцій, які ще не були записані в попередні блоки. Не всякий сформований блок буде прийнятий іншими учасниками. Потрібно, щоб числове значення хешу заголовка не перевищувало встановленого значення (параметр «складність»). Чим менше задано значення, тим менше ймовірність виконання умови. У службовій області блоку виділено місце для довільних значень. Якщо хеш заголовка незадовільний, довільні значення замінюються на нові і розрахунок хешу повторюється. Результат хешування (функції SHA-256) непередбачуваний, тому немає алгоритму цілеспрямованої зміни довільної області для досягнення бажаного результату. Зазвичай потрібна велика кількість перерахунків.

При формуванні блоків можуть виникнути ситуації, коли кілька нових блоків вважають попереднім один і той же блок. Це явище називається розгалуженням і відбувається через одночасне формування блоків «Майнер» (рис. 1.10).

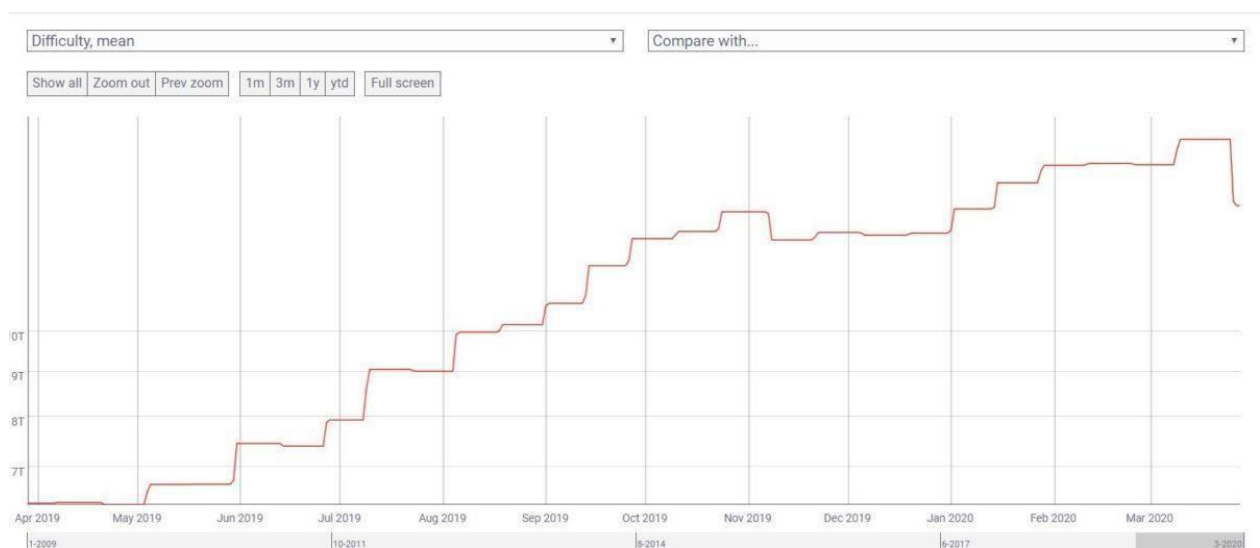


Рисунок 1.10 – Логарифмічний графік складності Майнінгу

До включення транзакції в блок є технічна можливість оформлення кількох різних транзакцій з передачі з однієї адреси одних і тих же біткоїнів різним одержувачам [36]. Як тільки транзакція буде включена в блок, інші транзакції з цими ж біткойнами система буде вже ігнорувати, тобто в ланцюжку блоків залишиться

тільки одна транзакція. Але якщо контролювати більше 50% сумарної обчислювальної потужності мережі, то існує теоретична можливість при будь-якому порозі підтверджень формувати паралельний більш довгий ланцюжок блоків, в якій ті ж біткоїни будуть передані іншому одержувачу [26] (проблема «подвійного витрачання»). Коли мережа отримає відомості про другий ланцюжок блоків, вона стане основною, а транзакція в ній – підтвердженою, перша ж транзакція втратить підтвердження і буде вважатися помилковою. В результаті не відбудеться подвоєння біткоїнів [36], але зміниться їх поточний власник, при цьому перший одержувач втратить біткоїни без будь-яких компенсацій.

Обмінний курс біткоїнів необхідно аналізувати з точки зору кількісної теорії грошей, яка є на сьогодні базовою в економічній науці. За цією теорією вартість грошей залежить всього від трьох параметрів:

$$\text{Вартість грошей} = Y / (V * M)$$

Y – це сума, на яку вироблено угод за певний період часу.

V – це швидкість обороту грошей.

M – це кількість грошей в обороті.

Традиційний підхід – враховувати тільки операції, які входять у Валовий внутрішній продукт (ВВП). А там враховуються угоди тільки з товарами призначеними для безпосереднього вживання. Не враховуються угоди з іншими валютами, сировиною і напівфабрикати, антикваріатом (то, що тільки змінює власника, але не проводиться) і т.д.

Відсутність монополії у виробництві біткоїнів – одна з найбільш привабливих його характеристик. Негативні економічні наслідки монополії відомі: товари і послуги, вироблені монополістом, коштують дорожче, ніж вироблені в умовах конкуренції, звичайно, якщо немає штучного обмеження цін. В результаті монополіст виробляє менший обсяг суспільного блага, а отримує за нього більше, ніж отримали б конкурентні виробники.

У сфері грошового обігу монополізм проявляється, як і в сфері товарного виробництва. Громадські витрати грошового обігу виявляються неефективно високими. Це проявляється:

по-перше, в тому, що держава привласнює сеньораж;

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		22

по-друге, виникають високі трансакційні витрати, які проявляються в великих банківських комісіях;

по-третє, грошова влада часто зловживає своїм правом грошової емісії. По суті, держава, випускаючи зайві паперові гроші, займається «псуванням» своєї національної валюти, що веде до високої інфляції і девальвації. Хвороби грошового обігу є однією з причин економічних криз.

Бар'єри, які споруджує державу на шляху грошового обігу, знижують ефективність економіки. Особливо великі ці бар'єри і витрати на шляху транскордонних грошових потоків, що проявляється не тільки в високих комісіях, а й в тривалості міжнародних переказів. Крім того, багато держав ввели обмеження на використання різних каналів грошового обігу. В результаті цього обмеження монопольну владу над багатьма з цих каналів отримали банки [17].

Економічні агенти відповіли на проблеми грошового обігу, що виникають з державної монополії в сфері грошової емісії, революційною інновацією – винаходом криптовалюти. Основною перевагою криптовалюти перед звичайною є істотно нижчі трансакційні витрати – витрати обігу – нульові або дуже низькі комісії. Трансакції, особливо транскордонні, займають значно менше часу, ніж у випадку використання звичайних валют.

В результаті розширення виробництва і використання криптовалюти держава може втратити свій важливий атрибут – право емісії національної валюти. Не виключено, що в доступному для огляду майбутньому національні валюти стануть атавізмом. Також і банки можуть втратити владу над каналами грошового обігу.

Позитивні властивості біткоїни

– У разі проведення комерційних операцій кожен учасник мережі може здійснювати миттєві операції з криптовалютою без посередників. Тобто покупець передає гроші безпосередньо продавцеві. Не потрібно йти в банк або вносити гроші на Ківі-гаманець, покупець або боржник просто відправляє біткоїни продавцеві або кредитуру. Монети в системі – це криптографічні (математичні) коди. Кожен з них абсолютно унікальний і не може використовуватися двічі.

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		23

- Біткоїни не залежить від інфляційних процесів. Їх не можна надрукувати, а створити величезну кількість одиниць не вийде. Кількість позначена на рівні програмного ходу.
- Відсутні посередники при проведенні маніпуляцій. Спеціально заблокувати переклад не зможе ні інша людина, ні організація.
- Незалежність. Безпека електронної готівки не залежить від місцезнаходження. Готівка можуть бути передана по комп'ютерних мережах.
- Безпека. Електронну готівку не можна скопіювати і повторно використовувати.
- Таємниця особистості. Таємниця особистості користувача захищена, зв'язок між користувачем і його покупками виявити неможливо.
- Автономний платіж. Коли користувач розплачується за покупку електронною готівкою, протокол між користувачем і продавцем виконується автономно. Тобто магазину не потрібно з'єднуватися з центральним комп'ютером для обробки платежу користувача.
- Подільність. Задана сума електронної готівки може бути поділена на частини меншої суми. (Звичайно, загальна сума в кінці повинна зійтися).
- Відкритий код криптовалюти. Вихідний код криптовалюти і теорія біткоїнів відкриті. У біткоїнів працюють ті ж алгоритми, які використовуються в інтернет-банкінгу. Єдиною відмінністю інтернет-банкінгу є розкриття інформації про кінцевого користувача. У мережі біткоїнів вся інформація про транзакції є в загальному доступі (скільки, коли), але немає даних про одержувача або відправника монет (немає доступу до персональної інформації власників гаманців). Однак у цього плюса є протилежний мінус.
- Відсутність інфляції. Кількість монет в цій системі зростає з певною швидкістю, закладеною таким чином, щоб вона збігалася зі швидкістю видобутку золота на планеті. Максимально можлива кількість монет строго обмежена і складає 21 мільйон біткоїнів. Так як немає ні політичних сил, ні корпорацій, здатних змінити такий порядок, в системі немає можливості для виникнення інфляції. Завдяки цьому, криптовалюта певною мірою є навіть надійніше золота [6]. Математичний розрахунок

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		24

гарантує дефіцит біткоїнів в недалекому майбутньому. Сьогодні кожен може стати володарем криптовалюти, яка ніколи не знеціниться, і буде тільки дорожчати. І це не порожня обіцянка, це математичне правило, яке закладено в кожній програмі-гаманці. Звичайно, його можна змінити, але за умови, що 99% власників гаманців погодяться з новими змінами.

– Пірінгова мережа криптовалюти. У пірінгових мережах немає головного сервера, що відповідає за всі операції. Протоколи працюють як однорангова мережа. Обмін інформацією (в нашому випадку – грошима) відбувається між 2-3 і більше програмами-клієнтами. Всі встановлені у користувачів програми-гаманці є частиною мережі біткоїни. Кожен клієнт зберігає запис про всі зроблені транзакції і про кількість біткоїнів на кожному гаманці [5]. Транзакції виробляються сотнями розподілених серверів, їх ще називають «здобувачами». Ні банки, ні податкові органи, ні держава не можуть контролювати обмін грошей між гаманцями користувачів.

– Безмежні можливості транзакцій. Кожен з власників гаманця може платити кому завгодно, де завгодно і за що завгодно. Транзакції неможливо проконтролювати або заборонити, так що можна здійснювати перекази в будь-яку точку світу, де б не перебував інший користувач з гаманцем біткоїнів. Криптовалюта біткоїни працює як «жива» готівка, поєднуючи в собі функції платіжного інструменту в електронній комерції. Не потрібно платити комісії і мита банкам і іншим організаціям. Третя сторона тут – математика, яка не враховує грошові аспекти системи. Комісійні в цій системі нижче, ніж в будь-який інший. Вони складають 0,1% від суми транзакції. Відсотки йдуть в гаманці «видобувачів» біткоїнів.

– Відсутність кордонів. У біткоїнів немає меж. Платежі, вчинені в цій системі, неможливо скасувати. Самі монети можна підробити, скопіювати або витратити двічі [29]. Такі можливості гарантує чесність всієї системи. Такі імениті бренди, як WordPress приймають біткоїни для оплати послуг. З кожним місяцем кількість інтернет-магазинів, ресурсів і компаній, що приймають до оплати біткоїни, зростає.

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		25

– Відсутність податків. При оплаті біткоїнів не стягується податків, оскільки офіційно біткоїни не є валютою. Це дає можливість придбати товар трохи дешевше.

Недоліки криптовалюти біткоїни

– Великі ризики. Ця нова віртуальна валюта нічим не забезпечена, окрім як попитом. Якщо попит впаде, то ціна на нього теж каменем піде вниз.

– Вона не легалізована. Є небезпека, що країни не стануть узаконювати цю валюту, а це сильно підірве до неї інтерес і багато хто не стане нею користуватися.

– Через відсутність контролю і анонімності платежів, нею легко можуть скористатися терористи, оплачуватися наркотики і т.д. Це створює велику проблему в узаконення біткоїнів і поки ця проблема ніяк не вирішена.

– Погана безпека. Відсутність належної безпеки від крадіжки, втрати та інші ризики. Оскільки все це поки не офіційно, то є ризик бути обманутим будь-якою системою обміну [4]. Також не варто забувати про загрози від вірусів, які можуть вкрасти доступ до гаманця і т.п. А повернути гроші буде неможливо, тому що в законах на даний момент не передбачені повернення віртуальних грошей.

– Сильний новинний вплив. Майже всі підйоми і падіння курсу біткоїнів безпосередньо залежали від оголошених заяв урядів різних країн. Висока волатильність курсу створює проблеми в короткостроковому періоді. Але, з іншого боку, якщо курс біткоїнів більш менш стабілізується і стане менш волатильним, то інвестиційний потенціал криптовалюти дуже сильно знизиться.

– Один з недоліків криптовалюти – висока схильність її курсу кон'юнктурних змін. На ринку явно «надувається міхур». Ціна біткоїнів б'є рекорди. Зрозуміло, що за злетом неминуче посідує падіння. Підвищена курсова нестабільність криптовалюти ускладнює їх використання в якості засобу обігу. Ця нестабільність пов'язана з відсутністю єдиного емісійного центру, зацікавленого в стабілізації курсу валюти. Таким центром в разі національної валюти є Центральний банк, який надає на національну валюту стабілізуючий вплив навіть в умовах політики вільного плавання.

Висновки до першого розділу

У рамках першого розділу даної дипломної роботи висвітлено теоретичні аспекти мобільного додатку «Гаманець для криптовалюти», на основі вищевикладеного варто зробити наступний висновок:

Розвиток Інтернету зробив істотний вплив на функціонування економіки багатьох держав. На його основі були створені мережі зв'язку, які суттєво підвищили швидкість передачі інформації, і тим самим сприяли прискоренню, укладення та виконання угод, підвищили швидкість обороту капіталу. Винахід передплачених карт, мобільних та інтернет-платежів привернув величезну кількість клієнтів в електронні грошові системи. Одним з останніх досягнень в цій області став винахід і розробка криптовалюти. Розробка і випуск криптовалюти залучили в Інтернет величезну кількість нових користувачів по всьому світу і число їх постійно збільшується.

Криптовалюта – це цифрова (віртуальна) валюта, одиниця якої – монета (англ. -Coin). Дану одиницю оплати неможливо підробити, так як вона складається з коду.

Криптовалюта представляє собою навмисно створену альтернативу валютним знаками вважається швидкими вірним способом оплати в мережі Інтернет. У неї немає жодного реального уявлення, тільки віртуальне. Це і відрізняє крипт валюту від інших віртуальних грошей. Вона створюється лише мережею. Пов'язана з технологією blockchain, що має на увазі ідею постійних записів всіх операцій з криптовалютою і даних її власників.

«Біткоїн» (англ. Bitcoin, від bit «біт» + coin «монета») – пірінгова платіжна система, яка використовує однойменну розрахункову одиницю – біткоїн.

Метою даної дипломної роботи є створення мобільного додатку «Гаманець для криптовалюти».

РОЗДІЛ 2

РОЗРОБКА МОБІЛЬНОГО ДОДАТКУ «ГАМАНЕЦЬ ДЛЯ КРИПТОВАЛЮТИ»

2.1 Аналіз і характеристика об'єкта проектування

Ціль розробки мобільного додатку «Гаманець для криптовалюти» реалізувати процес виводу інформації щодо балансу криптографічної валюти. На виході роботи мобільного додатку «Гаманець для криптовалюти» користувач отримує можливість поповнити власний рахунок а здійснити передачу криптографічних монет на зовнішню адресу.

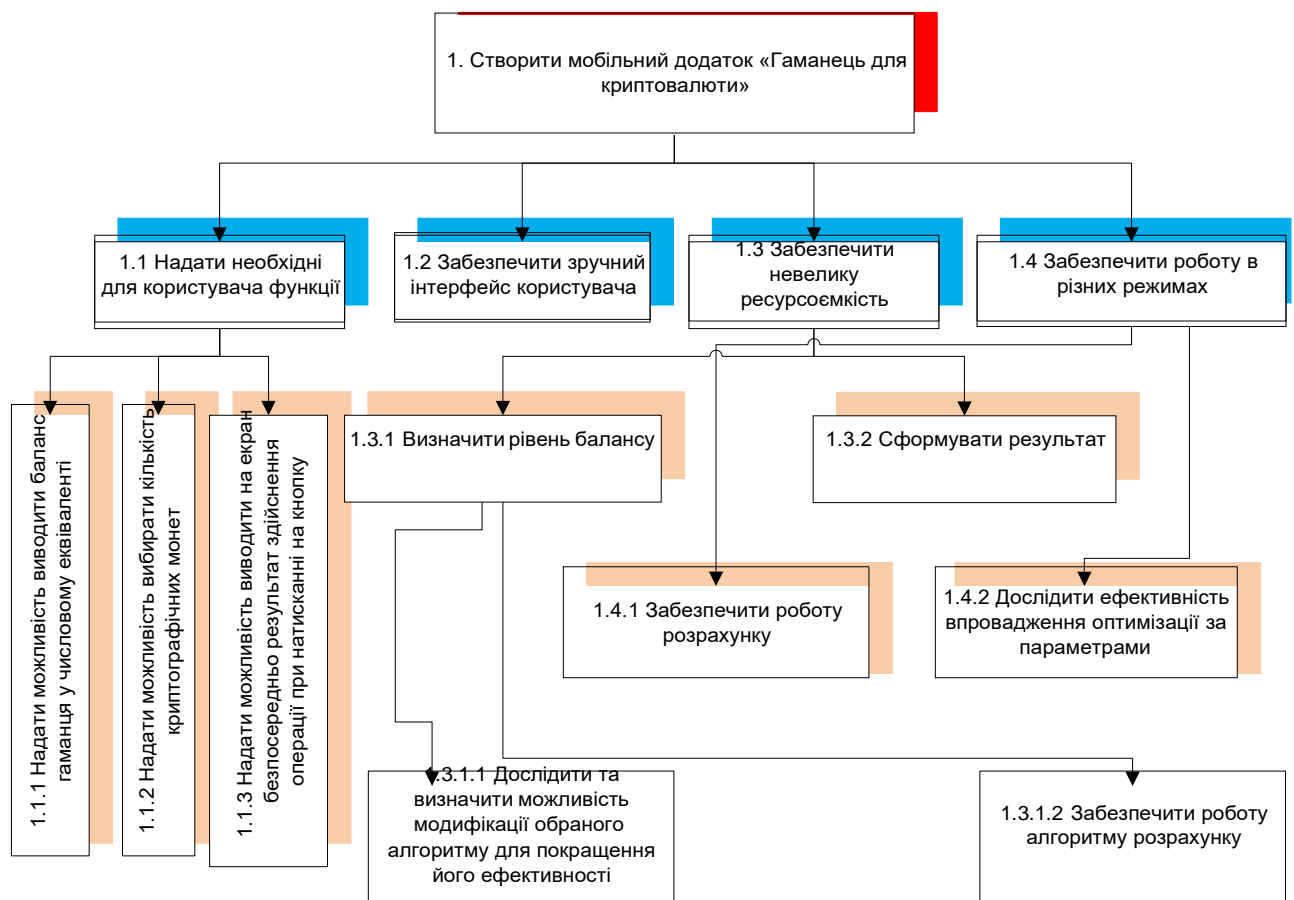


Рисунок 2.1 – Декомпозиція цілей

Вивід інформації планується тільки у текстовій формі з позначеннями операцій, які здійснено успішно або відхилено, таке рішення прийнято за рахунок того, що сучасний користувач більшою мірою сприймає зорову інформацію.

По результатах проведеного аналізу та беручи до уваги вищезазначені функції, було розроблено дерево цілей мобільного додатку «Гаманець для криптовалюти», що зображене на рис. 2.1. та дерево функцій які має виконувати мобільний додаток «Гаманець для криптовалюти»рис. 2.2.

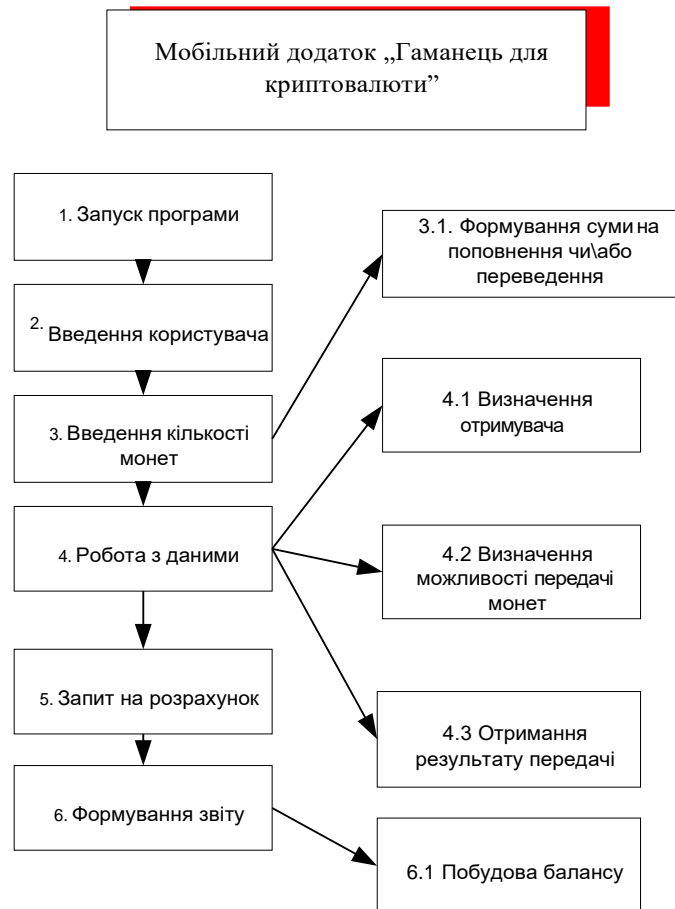


Рисунок 2.2 – Декомпозиція функцій

Відмінності створеного програмного продукту від аналогів полягають у наступному:

- додавання функцій окремого виводу результатів суми балансу;
- чіткий, яскравий коефіцієнт кольоровості;
- зменшення обчислювальних витрат – об’єм пам’яті та час виконання;
- спрощення обчислювальної структури;
- підвищення точності обчислень;

—покращення характеристик збіжності (час збіжності, а також конкретні величини, що характеризують збіжність);

—узгодження характеристик нового алгоритму з іншими процедурами, які використовуються разом з ним.

2.2 Інструментальні засоби розробки

Для розробки мобільного додатку «Гаманець для криптовалюти» використано мову програмування Swift та мобільну операційну систему iOS.

Swift – це швидка і ефективна мова програмування з відгуком в реальному часі, яку легко можна вставити в готовий код Objective-C.

Swift – це неймовірно потужна і проста мова програмування, представлена Apple в 2014 році, завдяки якій для багатьох програмістів відкривається можливість створення додатків для iOS, MacOS, Apple TV і Apple Watch. Дана мова успадкувала ідеї від Objective-C, Rust, Haskell, Ruby, Python, C# і CLU. Вона використовує патерни безпечного програмування і містить нові функції, що дозволяє зробити процес програмування захоплюючим і цікавим. Створений Крісом Латтнером, він вийшов надійною, швидкою і відкритою мовою, завдяки чому стало ще простіше створювати додатки для самого популярного телефону в світі.

Swift – це сучасна мова програмування зі стабільною релізною версією, яка досить проста в освоєнні, навіть якщо врахувати, що для багатьох це перша мова. Вона запобігає масі помилок на етапі компіляції, прискорює процес розробки і підвищує якість коду.

Основною філософією мови є її простота, швидкодія і захищеність. Поріг входження змінився в більшу сторону, в порівнянні з тим же Objective C – навіть не знаючи будь-яких інших мов програмування або основ, можна сміливо приступати до вивчення мови.

Apple максимально спростила мову для розробників: синтаксис став чистішим, що підвищує читаність коду як досвідченого, так і початківця програміста. Загальна кількість коду зменшилася, завдяки чому підвищилася продуктивність. Були повністю перероблені, переосмислені і переписані всі методи, які використовувалися в Objective C. Swift постійно розвивається і оновлюється, що говорить про її надійність і націленості на майбутнє.

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		30

Дану мову програмування розробники використовують для написання мобільних додатків для телефонів, розумних годинників і комп'ютерів компанії Apple [31].

Важливим моментом є те, що восени 2016 року Apple сприяла відкриттю центру навчання студентів, де кожен може вчитися програмуванню додатків для iOS. Але це ще не все, в штатах Swift вже включений в навчальну програму багатьох провідних навчальних закладів. У цієї мови програмування величезний потенціал.

На даний момент існує невелика кількість документації, в порівнянні з іншими мовами. Але це можна обумовити тим, що мова зовсім недавно побачила світ і стала Open Source проектом, адже на момент релізу його могли використовувати тільки ті розробники, у яких була активна підписка Apple Developer Program [12].

Ще раніше можна було сказати, що Swift не дуже популярний і для нього існує мала кількість сторонніх бібліотек і розширень, але зараз це зовсім не так. На GitHub і просторах інтернету знаходяться десятки відмінних рішень від сторонніх розробників, які допоможуть удосконалити Ваш проект.

У цієї сучасної й потужної мови програмування є ряд переваг, які виділяють її на тлі інших або того ж Objective-C. І це нові потужні мовні можливості, гранично сувора типізація даних, більш лаконічний синтаксис, істотно більш висока продуктивність в операціях доступу до колекцій (сортування і пошук), підтримка певної більшості сучасних синтаксичних конструкцій, що існують в інших високорівневих мовах програмування. І однією з головних особливостей Swift є те, що вона має повну зворотну сумісність з Objective-C. Таким чином це дозволяє поступово переписувати проекти з однієї мови на іншу, до того ж дає можливість використовувати функціонал Objective-C всередині самого Swift [22].

Одним з приємних моментів є Playground – це свого роду інтерактивна пісочниця, де розробник може написати код і відразу побачити результат своєї праці. Це дозволяє в кілька разів збільшити швидкість написання основного коду, адже написання окремих шматків коду або алгоритмів тепер можна проводити в плейграунді з візуальним зворотним зв'язком, без ризиків здійснення помилок в основному проекті.

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		31

Apple переробила безліч речей, завдяки чому ми отримали практично нову розвинену систему перерахування, ефективне управління аксесором і оптимізовану автоматичну збірку сміття.

Темною стороною мови є її невелика популярність і невелика поширеність. Більшість поточних проектів написані на Objective-C і їх доведеться підтримувати, та й багато компаній не переписують свої великі проекти тільки тому, що це величезний і неймовірно трудомісткий процес.

Apple зробила перехід від використання статичних бібліотек до динамічних і додала цей функціонал в свою мову програмування, що означає його більш швидкий розвиток, ніж самої iOS. Тепер розробники з легкістю можуть уявити свій додаток разом з написаними ними бібліотеками. Раніше було потрібно чекати нового релізу iOS, але тепер ситуація змінилася. Динамічні бібліотеки надають можливість для більш гнучких поліпшень і змін в коді, ніж будь-коли.

Swift більш функціональна і доступна мова, яку як і Objective-C підтримують офіційні фреймворки Cocoa і Cocoa Touch. У мережі є безліч відкритих проектів, в тому числі від таких великих компаній як Airbnb, які дозволяють використовувати всі сучасні методи і рішення при написанні коду.

Компілятор і стандартна бібліотека:

Swift: основний Swift репозиторій, який містить вихідний код для компілятора Swift, стандартна бібліотека і SourceKit;

Swift-Evolution: документи, що відносяться до триваючого розвитку Swift, включаючи цілі для майбутніх випусків, пропозиції для змін і розширень Swift;

Бібліотеки ядра:

Swift corelibs-foundation: вихідний код для Foundation, який надає загальну функціональність для всіх додатків;

Swift corelibs-libdispatch: вихідний код для libdispatch, який надає примітиви паралелізму для роботи на багатоядерному апаратному забезпеченні;

Swift corelibs-xctest: вихідний код для XCTest, який забезпечує фундаментальну інфраструктуру тестування для Swift-додатків і бібліотек;

Менеджер пакетів:

Swift package-manager: вихідний код для менеджера пакетів Swift;

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		32

Swift llbuild: вихідний код для llbuild, система низького рівня, яка використовує Swift package-manager;

Клоновані репозиторії:

Swift спирається на кілька інших проектів з відкритим кодом, особливо на компілятор LLVM.

Swift llvm: вихідний код LLVM, зі шматочками Swift-доповнень;

Swift clang: вихідний код для Clang, зі шматочками Swift-доповнень;

Swift lldb: вихідний код Swift-версії LLDB, для налагодження Swift програм.

iOS (до 24 червня 2010 року – iPhone OS) – мобільна операційна система для смартфонів, електронних планшетів, програвачів і деяких інших пристроїв, що розробляється і випускається американською компанією Apple. Була випущена в 2007 році; спочатку - для iPhone і iPod touch, пізніше – для таких пристроїв, як iPad і Apple TV. У 2014 році з'явилася підтримка автомобільних мультимедійних систем Apple CarPlay. На відміну від Windows Phone (Microsoft) і Android (Google), випускається тільки для пристроїв, вироблених фірмою Apple.

В iOS використовується ядро XNU, засноване на мікроядрі Mach і містить програмний код, розроблений компанією Apple, а також код з ОС NeXTSTEP і FreeBSD. Ядро iOS майже ідентично ядру настільної операційної системи Apple macOS (раніше називалася OS X). Починаючи з найпершої версії, iOS працює тільки на планшетних комп'ютерах і смартфонах з процесорами архітектури ARM.

Інтерфейс iOS заснований на концепції прямої взаємодії з використанням жестів «мультитач» [13]. Елементи управління інтерфейсом складаються з повзунків, перемикачів і кнопок.

iOS розроблена на основі операційної системи OS X (з тих пір перейменованої в macOS) і використовує той же набір основних компонентів Darwin, сумісний зі стандартом POSIX.

Шари абстракції iOS:

Core OS;

Core Services;

Media Layer;

Cocoa Touch

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		33

2.3 Алгоритм розробки мобільного додатку

Для реалізації мобільного додатку «Гаманець для криптовалюти» обрано каскадну модель життєвого циклу (рис. 2.3), яка реалізує, принцип одинарного виконання кожного з основних процесів і етапів в їх визначених рамках. Перехід на наступний етап реалізується після того, як буде проведена робота на поточному етапі, і відкатів на здійсненій стадії не передбачено. Кожен етап закінчується придбанням певного результату, який використовується в якості базової інформації для наступного етапу [16].



Рисунок 2.3 – Етапи розробки мобільного додатку «Гаманець для криптовалюти»

Розробка мобільних додатків складається з наступних функціональних виробничих етапів:

1. Розробка технічного завдання.

Приступити до реалізації проекту неможливо без ідеї, яка повинна пройти попереднє тестування. Тільки так можна буде зрозуміти, наскільки пропозиція відповідає споживчому попиту. Перш ніж приступити до розробки необхідно визначити обсяг цільової аудиторії і на підставі аналітичних робіт починати розробляти технічний план, який допоможе реалізувати проект якісним і функціональним.

2. Дизайн

Під кожний індивідуальний додаток необхідно придумати і розробити оригінальний дизайн. Зовнішнє і внутрішнє оформлення допомагає формувати довіру, лояльне ставлення і тим самим робить продукт більш успішним і конкурентоспроможним. Якісний дизайн надає унікальності додатку, тому актуальніше за все всі графічні і візуальні образи створювати «з нуля».

3. Створення копії або прототипу.

Така перевірочна робота дозволить наочно переконатися в тому, що додаток оснащено зручним і зрозумілим інтерфейсом, що він має всі якісні функціональні показники і працездатність ключових параметрів програми. Для програмування краще задіяти перевірені і сучасні технологічні рішення

4. Тестування готового продукту.

Експертна, перевірочна оцінка дозволить виключити всі недоліки і при необхідності провести деякі коригувальні процеси. Надалі залишається тільки знайти якісні і авторитетні магазини додатків. Публікація додатків на тематичних ресурсах дозволить ефективно розкручувати свій продукт, залучаючи максимальну кількість зацікавлених користувачів.

2.4 Архітектура мобільного «Гаманець для криптовалюти»

Загальна структура клієнт-серверної взаємодії з боку сервера представлена на рис. 2.4.

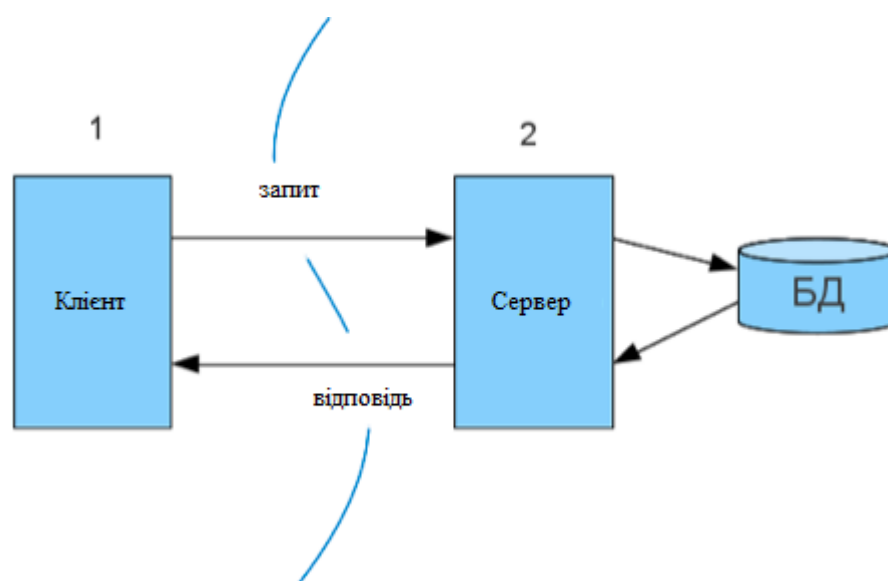


Рисунок 2.4 – Загальна структура клієнт-серверної взаємодії з боку сервера

Однак, у рамках роботи розглянемо цей же погляд з боку клієнта, і в зв'язку з цим, немає ніякої різниці між двуланковою і три ланковою архітектурою.

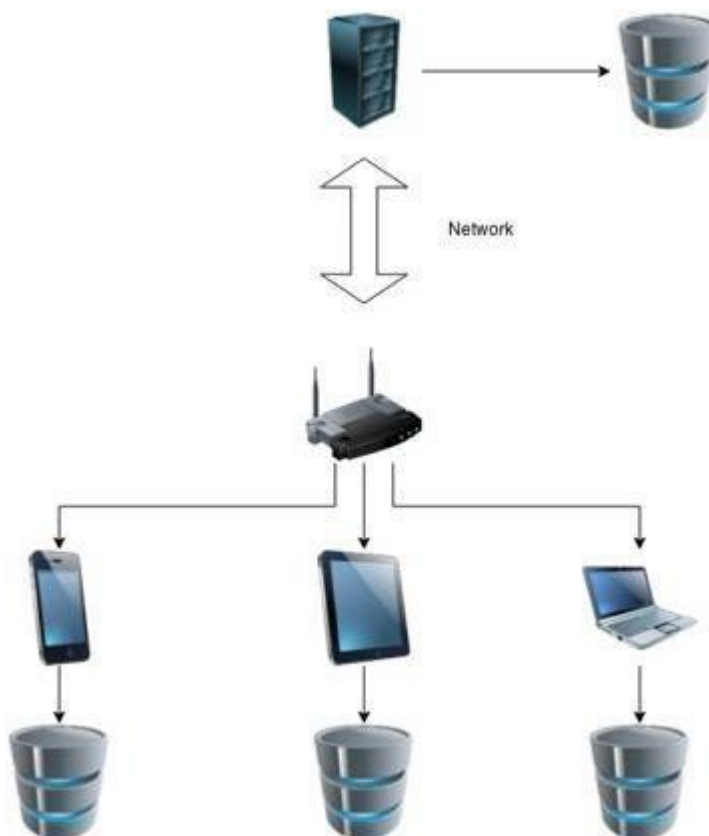


Рисунок 2.5 – Загальна структура програми

Може бути безліч клієнтів, які використовують один акаунт для спілкування з сервером.

Кожен клієнт, як правило, має своє власне локальне сховище. Таким чином, у ряді випадків, локальне сховище може бути синхронізоване з хмарою, і, відповідно, з кожним із клієнтів. Оскільки це окремий випадок і не впливає на архітектуру програми, опускаємо його [18].

Слід зазначити, що оскільки, деякі розробники прагнуть позбутися «серверної частини» деякі додатки побудовані навколо синхронізації їх сховищ в «хмарі». Тобто, фактично, мають так само, дволанкову систему, але з перенесенням архітектури її

розгортання на рівень операційної системи. У деяких випадках така структура виправдана, але така система не так легко масштабується, і її можливості дуже обмежені.

На самому примітивному рівні абстракції мобільний додаток складається з наступних архітектурних шарів:

Ядро додатку, яке включає в себе компоненти системи, не доступні для взаємодії з користувачем;

Графічний інтерфейс користувача;

Компоненти повторного використання: бібліотеки, візуальні компоненти й інше;

Файли оточення: AppDelegate, .plist і т. д.;

Ресурси програми: графічні файли, звуки, необхідні бінарні файли.

Найважливішою умовою побудови стійкої архітектури є відділення ядра системи від GUI, настільки, щоб одне, могло успішно функціонувати без іншого.

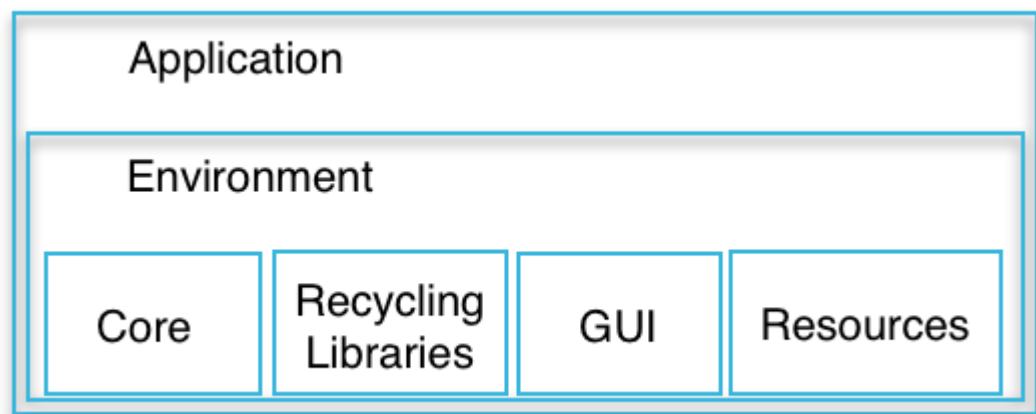


Рисунок 2.6 – Ядро додатку

Ядро додатку, складається з наступних шарів:

(Start layer) Стартовий шар, що визначає робочий процес, початку виконання програми.

(Network layer) Мережевий шар, що забезпечує механізм транспортної взаємодії.

(API layer) Шар API, що забезпечує єдину систему команд взаємодії між клієнтом і сервером.

(Network Cache Layer) Шар мережевого кешування, що забезпечує прискорення мережевої взаємодії клієнта і сервера.

(Validation Items Layer) Шар валідації даних отриманих з мережі.

(Network Items Layer) Шар суті даних переданих по мережі.

(Data Model) Модель даних, що забезпечує взаємодію сутностей даних.

(Local cache layer) Шар локального кешування, що забезпечує локальний доступ до вже отриманих мережевих ресурсів.

(Workflow layer) Шар робочих процесів, що включає класи і алгоритми специфічні для цього додатка.

(Local storage) Локальна пам'ять.

Одне з основних завдань, це забезпечити взаємно незалежне функціонування зазначених шарів. Кожен шар повинен забезпечувати тільки виконання покладених на нього функцій. Як правило, шар знаходиться на більш високому рівні ієрархії не повинен мати уявлення про специфіку реалізації інших шарів.

Постановка завдання дозволяє виділити кілька підзадач, які можуть бути описані окремими класами:

1) Завантаження даних з мережі.

Перевірка отриманих даних

Збереження даних в постійному сховищі.

Обчислення даних.

операція додавання

операція віднімання

Фільтрація даних по зазначеним критеріям (налаштування програми)

Клас старту програми.

2) Забезпечити пов'язану роботу інтерфейсу, який складається з наступних основних форм:

- головний контролер (може бути невидимим)
- форма обчислення

- сплеш
- опціональна форма налаштувань.

3) Після запуску програми на виконання, проводиться створення об'єкта, який відповідає за завантаження даних (в переважній більшості випадків асинхронних) і починає процес. Головний контролер додатку відображає сплеш-скрін, і в цей час, формує форму, яка займе його місце з приховування сплеш-форми.

4) Після закінчення завантаження даних, створюється об'єкт-валідатор і об'єкт-провайдер локального сховища. У разі якщо дані пройшли необхідну валідацію, вони можуть бути передані провайдеру локального сховища.

5) Для проведення обчислень створюється об'єкт-калькулятор, і об'єкти операцій. В об'єкт калькулятор передаються дані отримані з форми, і один з двох об'єктів операцій, котори знають як саме здійснити обчислення.

Звичайно, даний підхід вимагає більше зусиль по програмуванню, і відповідно, спочатку передбачає більше часу. Однак, виходячи з підзадач ясно, що по-перше, роботу над ним легко розпаралелити – в той час як один розробник зайнятий формуванням ядра – інший, створює і налагоджує UI.

Самі візуальні форми (GUI) відображають необхідний мінімум заснований на існуючих в ядрі завданнях.

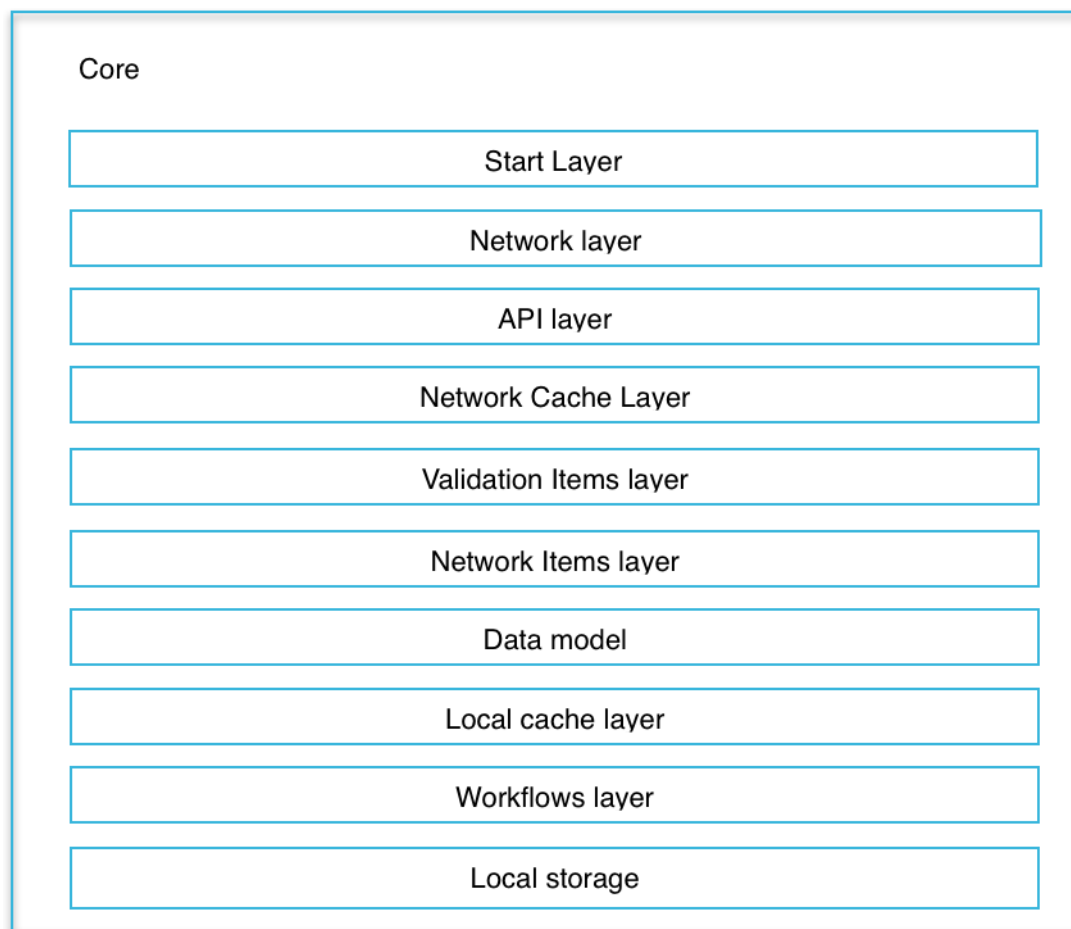


Рисунок 2.7 – Основні шари додатку

Start layer:

В iOS додаток починає функціонування з запуску об'єкта делегатного класу. Його призначення – прийняти і передати виклики системі з додатком, а також, здійснити первинну конфігурацію GUI додатку. Всі алгоритми і механізми, які не належать до старту програми, або отримання повідомлень від системи повинні бути винесені в окремі класи. Відразу після завершення початкової конфігурації управління повинно бути передано класу, який здійснює інші операції налаштування програми: авторизацію, реконфігурування інтерфейсу в залежності від умов, початкове завантаження даних, отримання необхідних токенів і так далі.

Network Layer:

Забезпечує базові алгоритми транспортного рівня передачі повідомлень від клієнта до сервера, і отримання від нього необхідної інформації. Як правило, повідомлення можуть передаватися в форматах JSON і Multipart, хоча, в деяких екзотичних випадках це може бути XML або взагалі бінарний потік. Крім того, кожне

повідомлення може мати заголовок зі службовою інформацією. Наприклад, там може бути описана тривалість зберігання запиту / відповіді в кеші програми.

Network Layer не має жодного уявлення про використовувані додатком сервери, або про його системи команд. Обробка помилок з'єднання з мережею здійснюється віртуальними методами на наступних рівнях додатку. Завдання цього шару тільки здійснити виклик методу обробки і передати в нього отриману з мережі інформацію.

Крім того, перед безпосереднім запитом інформації з мережі, network layer опитує локальний кеш, і в разі присутності там відповіді відразу ж повертає його користувачеві.

Socket – найбільш низькорівневий підхід, що включає в себе синхронні і асинхронні запити, і має можливість працювати як з TCP так і з UDP підключеннями. Дозволяє робити практично все що завгодно, але вимагає високого ступеня концентрації на завданні.

WebSocket – підхід спирається на використання заголовків поверх TCP.

WCF – напевно найдосконаліший механізм, але він має настільки серйозний мінус, який переважає всі плюси. Підхід винайдений в надрах Microsoft спирається на створенні проксі-класу, який опосередковує взаємовідношення між логікою програми, і віддаленим сервером.

REST – надійний, перевірений часом компроміс всіх перерахованих вище підходів. Звичайно, від частини повноважень кожного з підходу доводиться відмовлятися, зате робиться це швидко, і надзвичайно ефективно з мінімумом зусиль.

API Layer:

Описує команди REST і здійснює вибір хосту. API Layer повністю відділений від знання реалізації мережевих протоколів і будь-яких інших особливостей функціонування програми. Технічно, він може бути повністю замінений, без будь-яких змін в інших частинах програми.

Клас успадкований від ClientBase. Код класу настільки простий, що немає необхідності навіть приводити його цілком – він складається їх однакового опису API

Network Cache Layer:

Даний шар кешування задіюється для прискорення мережевого обміну між клієнтом і сервером на рівні iOS SDK. Вибір відповідей здійснюється стороною, що

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		41

лежить за межами контролю системи, і не гарантує зниження мережевого трафіку, але прискорює його. Доступу до даних або механізмів реалізації немає ні з боку додатка, ні з боку системи. При цьому використовується SQLite сховище.

Код необхідний для цього занадто простий, щоб не використовувати його в будь-якому проекті, який має доступ до мережі.

Validation Items layer:

Формат одержуваних даних з мережі більшою мірою залежить від розробників сервера. Додаток фізично не може контролювати використання спочатку заданого формату. Для складно-структурованих даних, корекція помилок порівнянна по складності з розробкою самого додатка. Наявність помилок, в свою чергу, може призвести до креш додатку. Використання механізму валідації даних істотно знижує загрозу некоректної поведінки. Шар валідації складається з схем JSON для більшості запитів до сервера, і класу, який здійснює перевірку отриманих даних на відповідність завантаженої схеми.



Рисунок 2.8 – Local cache layer

Якщо отриманий пакет не відповідає схемі, він відхиляється додатком. Зухвалий код отримає повідомлення про помилку. Аналогічне повідомлення буде записано в лог консолі. Крім того, може бути викликана команда сервера для передачі на сторону сервера звіту, про виниклу помилку. Головне, передбачити вихід з рекурсії, якщо команда відправки такого повідомлення теж викликає якусь помилку (4xx або 5xx).

Network Items layer:

Саме на цьому шарі лежить відповідальність за мапінг даних з JSON в десеріалізоване уявлення. Даний шар використовується для опису класів, які здійснюють об'єктне або об'єктно-реляційне перетворення. У мережі існує велика кількість бібліотек, які здійснюють об'єктно-реляційне перетворення.

Local cache layer:

До завдань даного шару відносяться:

Кешування завантажуваних з мережі зображень.

Кешування запитів / відповідей сервера

Формування черги запитів в разі відсутності мережі і роботи користувача оффлан.

Моніторинг кеш даних і очищення даних, термін життя яких закінчився.

Повідомлення додатку про неможливість отримати інформацію щодо заданого об'єкту з мережі.

Workflows layer:

Всі реалізовані алгоритми, які не належать до верствам ядра, і не являють собою GUI повинні бути винесені в класи специфічних послідовностей робочих процесів. Кожен з цих процесів оформляється в своєму стилі, і підключається до основної частини програми шляхом додавання посилань на екземпляр відповідного класу в GUI. У переважній більшості випадків, всі ці процеси не є візуальними. Однак є деякі винятки, наприклад, коли необхідно здійснити довгу послідовність визначених кадрів анімації, із заданими алгоритмами відображення

Кожен виклик шарів ядра супроводжується передачею об'єкта зворотного виклику (callback), і саме через нього повинно бути повернуто управління в додаток при успішному виконанні команди або виникненні помилок.

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		43

Використання NSDictionary до якості формату отриманих даних дозволяє автоматично вирішити ще цілий ряд архітектурних проблем:

дані в масивах можуть бути представлені точно в тій послідовності, в якій вони були отримані від сервера.

дані однозначно відповідають вибраному запиту до сервера, з точністю до переданих параметрів в POST запиті (тобто легко було відрізнити об'єкт отриманий від певної команди, від об'єкта отриманого від тієї ж команди, але з іншими даними, переданими в якості параметрів POST пост запиту).

Атомарність запису об'єкта даних в постійне сховище.

Миттєвість і атомарність читання даних з постійного сховища.

Відсутність необхідності в нормалізації даних.

Незалежність в інтерпретації даних.

Всі дані завжди актуальні.

Обсяг коду підтримки мінімальний (1 рядок).

Засоби читання / запису iOS SDK робить NSDictionary ідеальним форматом збереження відносно невеликих короткоживучих даних, оскільки для цього застосовуються однопрохідні алгоритми.

Для читання серіалізованих збережених даних немає необхідності в тому, щоб задіяти додаткову логіку. Повернення даних може здійснюватися тією ж командою, яка читає дані з мережі.

Негативною стороною такого підходу вважається, що це погано впливає на продуктивність пристрою, однак, вивчення питання показує, що обсяг таких даних не перевищує 5Кбайт, дані завантажуються в пам'ять миттєво, єдиним блоком, і таким же чином звільняються з пам'яті, відразу ж після того, як в них відпадає необхідність, наприклад, коли ViewController перестає існувати. У той же час читання даних блоками (підрядник) з бази даних SQL породжує велику кількість об'єктів (на рівні, що виходить за рамки контролю додатку), які сумарно перевищують зазначений обсяг, до того ж, створюють додаткове навантаження на процесор. Використання центрального сховища виправдано тоді, коли дані повинні зберігатися довгий час, протягом багатьох сесій роботи програми. При цьому, дані з мережі завантажуються частково.

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		44

Висновки до другого розділу

У межах другого розділу даної дипломної роботи здійснено розробку мобільного додатку «Гаманець для криптовалюти», наведено функціональні особливості та інструментальні засоби.

Для розробки мобільного додатку «Гаманець для криптовалюти» використано мову програмування Swift.

Для реалізації мобільного додатку «Гаманець для криптовалюти» обрано каскадну модель життєвого циклу, яка реалізує, принцип одинарного виконання кожного з основних процесів і етапів в їх визначених рамках. Перехід на наступний етап реалізується після того, як буде проведена робота на поточному етапі, і відкатів на здійсненій стадії не передбачено. Кожен етап закінчується придбанням певного результату, який використовується в якості базової інформації для наступного етапу.

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		45

РОЗДІЛ 3

ПРОЕКТУВАННЯ МОБІЛЬНОГО ДОДАТКУ «ГАМАНЕЦЬ ДЛЯ КРИПТОВАЛЮТИ»

3.1 Структура мобільного додатку «Гаманець для криптовалют»

Формування основних

*Begin PBXGroup section */*

```
046F9946220E10FB005A46BB = {
    isa = PBXGroup;
    children = (
        046F9951220E10FB005A46BB /* CryptoBTC */,
        046F9950220E10FB005A46BB /* Products */,
    );
    sourceTree = "<group>";
};
046F9950220E10FB005A46BB /* Products */ = {
    isa = PBXGroup;
    children = (
        046F994F220E10FB005A46BB /* CryptoBTC.app */,
    );
    name = Products;
    sourceTree = "<group>";
};
046F9951220E10FB005A46BB /* CryptoBTC */ = {
    isa = PBXGroup;
    children = (
        046F9952220E10FB005A46BB /* AppDelegate.swift */,
        046F9954220E10FB005A46BB /* ViewController.swift */,
        046F9956220E10FB005A46BB /* Main.storyboard */,
        046F9959220E10FC005A46BB /* Assets.xcassets */,
        046F995B220E10FC005A46BB /*
LaunchScreen.storyboard */,
        046F995E220E10FC005A46BB /* Info.plist */,
        046F9964220E1536005A46BB /* SendController.swift */,
    );
    path = CryptoBTC;
    sourceTree = "<group>";
};
```

Функція переведення монет описано як:

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		46

```

func setupAlert(title : String)
{
    let alert = UIAlertController(title: title, message: "", preferredStyle: .alert)
    alert.addAction(UIAlertAction(title: "OK", style: .default, handler: { (action) in
        self.navigationController?.popViewController(animated: true)
    }))
    present(alert, animated: true, completion: nil)
}
func checkSum(Second : Int)
{
    if defaults >= Second
    {
        defaults -= Second
        save.set(defaults, forKey: "money")
        setupAlert(title: "Успешно")
    } else {
        setupAlert(title: "У вас не хватает BTC")
    }
}

```

Процедура виведення балансу задається як:

```

@IBOutlet weak var sendBtn: UIButton!
@IBOutlet weak var rechargeBtn: UIButton!
override func viewDidLoad() {
    super.viewDidLoad()
    setupBtn()
    title = "Мой баланс"
    if defaults.integer(forKey: "money") > 0
    {
        money += defaults.integer(forKey: "money")
    } else { money = 1000}
    navigationItem.backBarButtonItem = UIBarButtonItem()
    defaults.set(money, forKey: "money")
}

```

Функція поповнення монетами гаманця описується як:

```

}
override func viewWillAppear(_ animated: Bool) {
    money = defaults.integer(forKey: "money")
}
func setupBtn()
{
    sendBtn.layer.cornerRadius = 8
    rechargeBtn.layer.cornerRadius = 8
}
@IBAction func sendBtc(_ sender: Any) {

```

```

}
@IBAction func rechargeBtc(_ sender: Any) {
    money += 1000
    defaults.set(money, forKey: "money")
}

```

3.2 Інструкція роботи користувача з системою

Тестування розробленого мобільного додатку «Гаманець для криптовалюти» здійснимо на найбільш навантаженому режимі.

Отримані результати наведено на рис. 3.1 – 3.10.

На рис. 3.1 наведено мобільний телефон без додатку. Здійснюємо компіляцію (рис. 3.2) и на екрані з'являється іконка розробленого мобільного додатку (рис. 3.3).

Наступним кроком запускаємо розроблений мобільний додаток та на екрані бачимо головне вікно програми (рис. 3.4).

Далі проводимо тестування розробленого мобільного додатку.



Рисунок 3.1 – Загальний вигляд екрану мобільного телефону

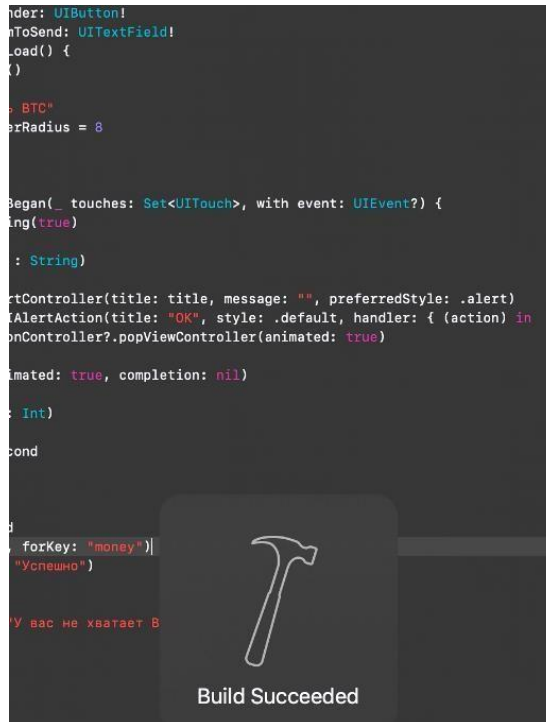


Рисунок 3.2 – Процес компіляції



Рисунок 3.3 – Іконка розробленого мобільного додатку

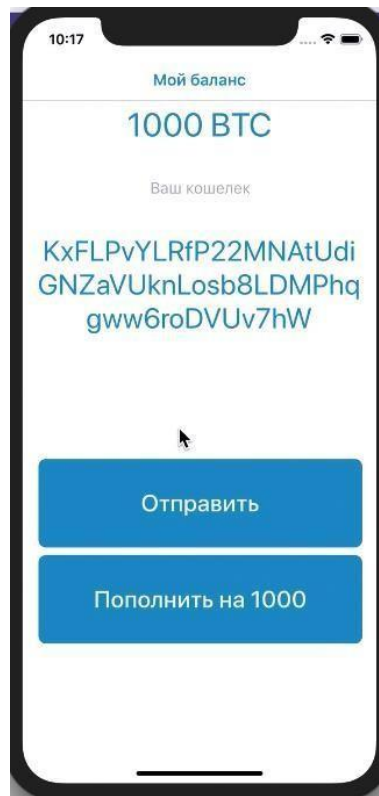


Рисунок 3.4 – Головне вікно розробленого мобільного додатку

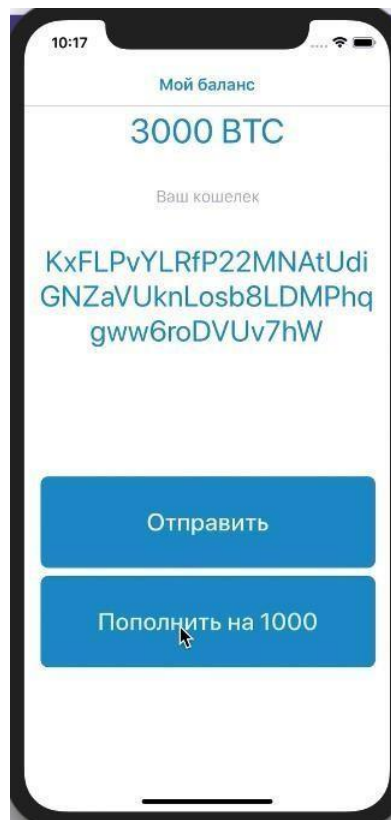


Рисунок 3.5 – Тестування розробленого додатку

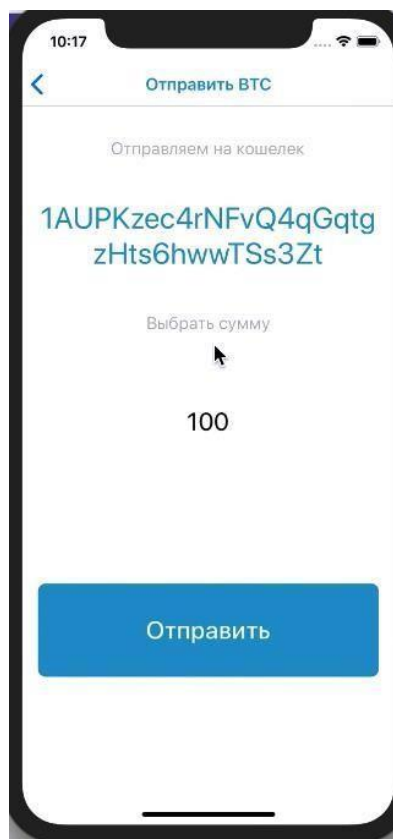


Рисунок 3.6 – Тестування розробленого додатку

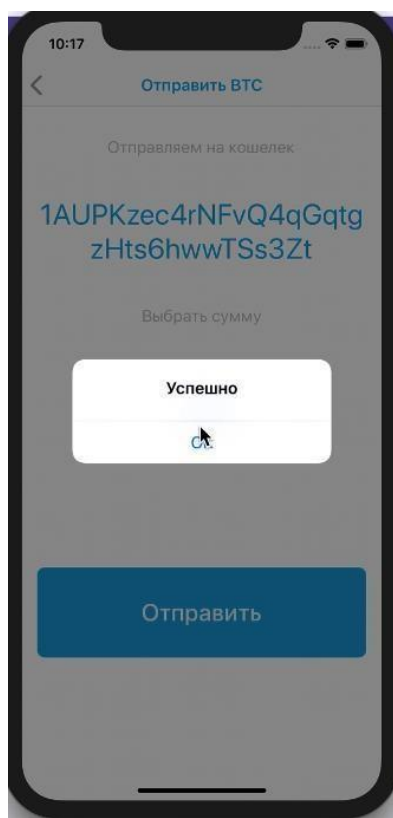


Рисунок 3.7 – Тестування розробленого додатку

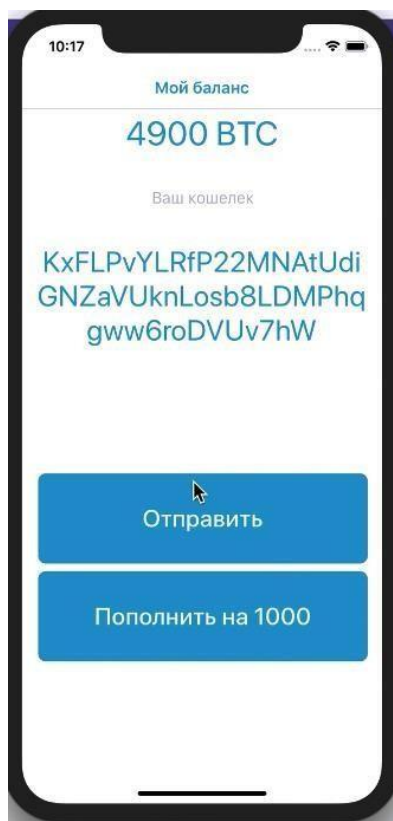


Рисунок 3.8 – Тестування розробленого додатку



Рисунок 3.9 – Тестування розробленого додатку

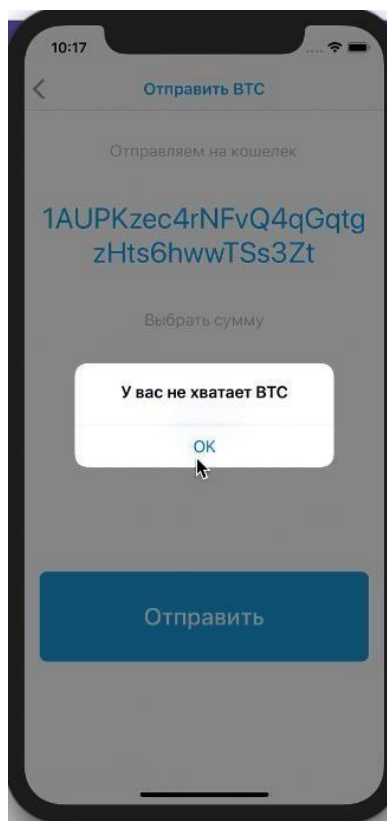


Рисунок 3.10 – Тестування розробленого додатку

Під час тестування збоїв та недоліків у роботі мобільного додатку не виявлено, що говорить про високу якість розробки та можливість встановлення за потребою на телефони або планшети з мобільною інформаційною системою iOS.

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		53

Висновки до третього розділу

У межах третього розділу здійснено тестування розробленого мобільного додатку «Гаманець для криптовалюти».

Під час тестування збоїв та недоліків у роботі мобільного додатку не виявлено, що говорить про високу якість розробки та можливість встановлення за потребою на телефони або планшети з мобільною інформаційною системою iOS.

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		54

ВИСНОВКИ

У рамках даної дипломної роботи здійснено створення мобільного додатку «Гаманець для криптовалюти».

На сторінках дослідження:

- розкрито функціональні особливості мобільного додатку «Гаманець для криптовалюти»;
- описано інструментальні засоби розробки;
- визначено методологію розробки мобільного додатку «Гаманець для криптовалюти»;
- досліджено архітектуру мобільного додатку «Гаманець для криптовалюти»;
- наведено структуру мобільного додатку «Гаманець для криптовалюти»;
- проведено тестування розробленого мобільного додатку «Гаманець для криптовалюти»;
- запропоновано верифікацію результатів дослідження.

Розвиток Інтернету зробив істотний вплив на функціонування економіки багатьох держав. На його основі були створені мережі зв'язку, які суттєво підвищили швидкість передачі інформації, і тим самим сприяли прискоренню, укладення та виконання угод, підвищили швидкість обороту капіталу. Винахід передплачених карт, мобільних та інтернет-платежів привернув величезну кількість клієнтів в електронні грошові системи. Одним з останніх досягнень в цій області став винахід і розробка криптовалюти. Розробка і випуск криптовалюти залучили в Інтернет величезну кількість нових користувачів по всьому світу і число їх постійно збільшується.

Криптовалюта – це цифрова (віртуальна) валюта, одиниця якої – монета (англ. -Coin). Дану одиницю оплати неможливо підробити, так як вона складається з коду.

Криптовалюта представляє собою навмисно створену альтернативу валютним знаками вважається швидкими вірним способом оплати в мережі Інтернет. У неї немає жодного реального уявлення, тільки віртуальне. Це і відрізняє крипт валюту від інших віртуальних грошей. Вона створюється лише мережею. Пов'язана з

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		55

технологією blockchain, що має на увазі ідею постійних записів всіх операцій з криптовалютою і даних її власників.

«Біткоїн» (англ. Bitcoin, від bit «біт» + coin «монета») – пірінгова платіжна система, яка використовує однойменну розрахункову одиницю – біткоїн.

Для розробки мобільного додатку «Гаманець для криптовалюти» використано мову програмування Swift.

Для реалізації мобільного додатку «Гаманець для криптовалюти» обрано каскадну модель життєвого циклу, яка реалізує, принцип одинарного виконання кожного з основних процесів і етапів в їх визначених рамках. Перехід на наступний етап реалізується після того, як буде проведена робота на поточному етапі, і відкатів на здійсненій стадії не передбачено. Кожен етап закінчується придбанням певного результату, який використовується в якості базової інформації для наступного етапу.

Під час тестування збоїв та недоліків у роботі мобільного додатку не виявлено, що говорить про високу якість розробки та можливість встановлення за потребою на телефони або планшети з мобільною інформаційною системою iOS.

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		56

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Апалькова В. В. Розвиток ринку міжнародних платіжних послуг в контексті глобальної інформатизації: автореф. дис. ... канд. екон. наук: 08.00.02 / В. В. Апалькова; МОН України. – Київ. нац. ун-т. ім. Тараса Шевченко. – Київ, 2016. – 23 с.
2. Банковских сотрудников ждут массовые сокращения [Електронний ресурс]. – Режим доступу: <http://psm7.com/news/bankovskix-sotrudnikov-zhdut-massovye-sokrashheniya.html>.
3. Вігірінський Андрій. Розвиток безготівкових розрахунків - подарунок комерційним банкам від НБУ вартістю в \$1,5 млрд. / Андрій Вігірінський [Електронний ресурс]. – Режим доступу: <http://uainfo.org/blognews/1439302650-rozvitok-bezgotivkovih-rozrahunkiv---podarunok-komertsiynim.html>
4. Войт Н. Н. Информатика и вычислительная техника. – Ульяновск: УлГТУ 2013. – 362 с.
5. Волосович С. В. Інноваційні платіжні послуги як інструмент розвитку торгівлі / С. В. Волосович // Економічний вісник університету. – 2016. – № 29/1. – С. 12–18.
6. Голощапов А. А. Google Android: программирование для мобильных устройств. – Спб.: БХВ – Петербург 2014. – 163 с.
7. Громова Тетяна. Кто съест мировые банки / Татьяна Громова // Власть денег. – 2016. – № 5 (442) [Електронний ресурс]. – Режим доступу: <http://www.dsnews.ua/future/ktosest-mirovye-banki-04052016111700>.
8. Діордієв В. О. Засади створення фінансового інноваційного хабу в Україні / В. О. Діордієв [Електронний ресурс]. – Режим доступу: <http://www.m.nayka.com.ua/?op=1&j=efektyvna-ekonomika&s=ua&z=5207>.
9. Европейцы воспользуются всеми преимуществами единого цифрового рынка [Електронний ресурс]. – Режим доступу: <http://psm7.com/evropejcy-vospolzuyutsya-vsemipreimushhestvami-edinogo-cifrovogo-rynka.html>

10. Зеньков Алексей. Почему финтех сейчас – уже больше, чем просто финтех / Алексей Зеньков [Электронный ресурс]. – Режим доступа: <http://rb.ru/story/not-just-fintech/>

11. Інновації майбутнього: FinTech Cluster, розвиток Bitcoin в Україні, а також «реінкарнація» НСМЕП [Електронний ресурс]. – Режим доступу: <http://news.finance.ua/ua/news/-/352397/innovatsiyi-majbutnogo-fintech-cluster-rozvytok-bitcoin-v-ukrayini-atakoz-reinkarnatsiya-nsmep>. Актуальні проблеми міжнародних відносин. Випуск 131. 2017 105

12. Комплексна програма розвитку фінансового сектору України до 2020 року [Електронний ресурс]. – Режим доступу: <https://bank.gov.ua/doccatalog/document?id=18563297>.

13. Лекция 1: Мобильные устройства и их характеристики. [Электронный ресурс] URL: Электронный научный журнал «Наука и перспективы» №3, 2016 <http://www.intuit.ru/studies/courses/574/430/lecture/9745> (дата обращения 20.04.2020)

14. Очимовская Татьяна. Цифровая революция добьет банки / Власть денег. – 2015 – № 5 // Татьяна Очимовская [Электронный ресурс]. – Режим доступа: <http://www.dsnews.ua/economics/tsifrovaya-revolutsiya-dobet-banki-08052015185300>.

15. Пискунова Н. В. Заработать миллионы с Iphone и Android пользователей. – М.: Финансы и статистика 2015. – 162с.

16. Плескач В. Л. Моделювання фінансово-економічних процесів: монографія / В. Л. Плескач. – К.: Київ. нац. торг.-екон. ун-т, 2010. – 428 с.

17. Пол Винья и Майкл Кейси. Эпоха криптовалют. Как биткоин и блокчейн меняют мировой экономический порядок – М.: Манн, Иванов и Фербер, 2017. – 432 с.

18. Проблеми правового регулювання відносин на ринку фінансових послуг в умовах глобалізації : монографія / Чернадчук В.Д., Дерев'янко Б.В., Старинський М.В., Афанасієв Р.В., Гончарова А.В., Маланчук Т.В., Плотнікова М.В., Швагер О.А.; за заг. ред. В.Д. Чернадчука, М.В. Старинського. – Суми : видавничо-виробниче підприємство «Мрія», 2017. – 232 с.

19. Размывание границ: Как компании сегмента FinTech влияют на сектор финансовых услуг. Всемирный обзор сегмента FinTech. – Март 2016 г.[Электронный

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		58

ресурс]. – Режим доступа: <http://www.pwc.ru/ru/banking/publications/fintech-global-report-rus.pdf>.

20. Сергей Базанов. Биткоин для всех. Популярно о первой распределенной одноранговой денежной системе. – Ridero, 2018. – 248 с.

21. Соколов В. В. Вычислительная техника и информационные технологии. Разработка мобильных приложений. Учебное пособие. – М.: Юрайт 2016. – 176 с.

22. Тенденции и перспективы развития банкинга: 2016 и 2017 [Электронный ресурс]. – Режим доступа: <http://psm7.com/news/tendencii-i-perspektivy-razvitiya-bankinga-2016-i-2017.html>.

23. Управление – Экскурсия по мировым финтех-хабам [Электронный ресурс]. – Режим доступа: <https://habrahabr.ru/company/payonline/blog/305570>.

24. Циганов С. А. Структурні зміни на глобальному ринку електронних платіжних по- слуг / С. А. Циганов, В. В. Апалькова // Інвестиції: практика та досвід. – 2016. – № 3. – С. 26–33.

25. Bannister David. Payments innovation: lessons from emerging markets / David Bannister [Electronic Resource]. – Mode of access: <http://www.bankingtech.com/294492/paymentsinnovation-lessons-from-emerging-markets>.

26. Bourque Andre. 12 Top Fintech Companies to Watch / Andre Bourque [Electronic Resource]. – Mode of access: <https://www.entrepreneur.com/article/288334>. 106 Actual problems of international relations. Release 131. – 2017

27. Dale P., Morgan Hein iOS для программистов. Создаем приложения. – СПб.: Питер 2012. – 560 с.

28. Degeler Andrii. How the W3C is trying to standardize payments on the internet / Andrii Degeler [Electronic Resource]. – Mode of access: <http://thenextweb.com/dd/2015/08/02/w3c-trying-standardize-payments-internet>

29. Dickerson James. The Future of Fintech and Banking: Digitally disrupted or reimagined? – 12 p. / James Dickerson, Samad Masood, Julian Skan [Electronic Resource]. – Mode of access: <http://www.fintechinnovationlablondon.co.uk/media/730274/Accenture-The-Future-of-Fintech-and-Banking-digitallydisrupted-or-reima-.pdf>.

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		59

30. Digital Agenda for Europe [Electronic Resource]. – Mode of access: <http://ec.europa.eu/digitalagenda>.
31. Ernst & Young Global Limited, «Блокчейн: цепная реакция» 2016.
32. Europa–2020. A European strategy for smart, sustainable and inclusive growth [Electronic Resource]. – Mode of access: <http://ec.europa.eu/eu2020/pdf/COMPLET%20EN%20BARROSO%20%20%20007%20-%20Europe%202020%20-%20EN%20version.pdf>.
33. Global B2C cross-border ecommerce to skyrocket: USD 1 trillion in 2020 [Electronic Resource]. – Mode of access: <http://thepaypers.com/ecommerce/global-b2c-cross-borderecommerce-to-skyrocket-usd-1-trillion-in-2020/760244-25>
34. Innovate Finance Global Summit 2017 [Electronic Resource]. – Mode of access: <http://ifgs.innovatefinance.com>
35. Murata С. Империя приложений. – М.: Альпина Паблишер 2013. – 236 с.
36. Nielsen J., Budiu R. Mobile Usability. – Спб.: Эксмо 2013. – 256 с
37. Payment Services Directive and Interchange fees Regulation: frequently asked questions. EUROPEAN COMMISSION. – Brussels. – 2013. – 24 July [Electronic Resource]. – Mode of access: [http://europa.eu/rapid/press-release MEMO-13-719_en.htm](http://europa.eu/rapid/press-release_MEMO-13-719_en.htm)
38. Paytm and Transmit Security win FT Fintech awards [Electronic Resource]. – Mode of access: <https://www.ft.com/content/1a92d762-ac1b-11e6-ba7d-76378e4fef24>
39. Rennick Emmet. The Fintech 2.0 paper. Rebooting Financial Services / Emmet Rennick [Electronic Resource]. – Mode of access: <http://www.oliverwyman.com/insights/publications/2015/jun/the-fintech-2-0-paper.html#.V-VYm4iLS71>
40. Saifedean Ammous. The Bitcoin Standard: The Decentralized Alternative to Central Banking. – New York: John Wiley & Sons Inc, 2018. – 304 с.
41. Secret of Building a Successful FinTech Startup in 2016 [Part 1] [Electronic Resource]. – Mode of access: <https://letstalkpayments.com/secret-of-building-a-successful-fintechstartup-in-2016-part-1>
42. Stevens С. Миллионеры из AppStore. Секреты разработчиков приложений бесцеллеров. – М.: Манн, Иванов и Фербер 2012. – 256 с.

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		60

43. The Rise of FinTech in China. Redefining Financial Services [Electronic Resource]. – Mode of access: [http://www.ey.com/Publication/vwLUAssets/ey-the-rise-of-fintech-in-china/ \\$FILE/ey-the-rise-of-fintech-in-china.pdf](http://www.ey.com/Publication/vwLUAssets/ey-the-rise-of-fintech-in-china/$FILE/ey-the-rise-of-fintech-in-china.pdf)

44. The Role of Financial Services in Society Understanding the impact of technology – enabled innovation on financial stability. WORLD ECONOMIC FORUM, 2016 – 16 p. [Electronic Resource]. – Mode of access: http://www3.weforum.org/docs/WEF_FS_Role_FinancialServicesSociety_Stability_Tech_Recommendations_2016.pdf

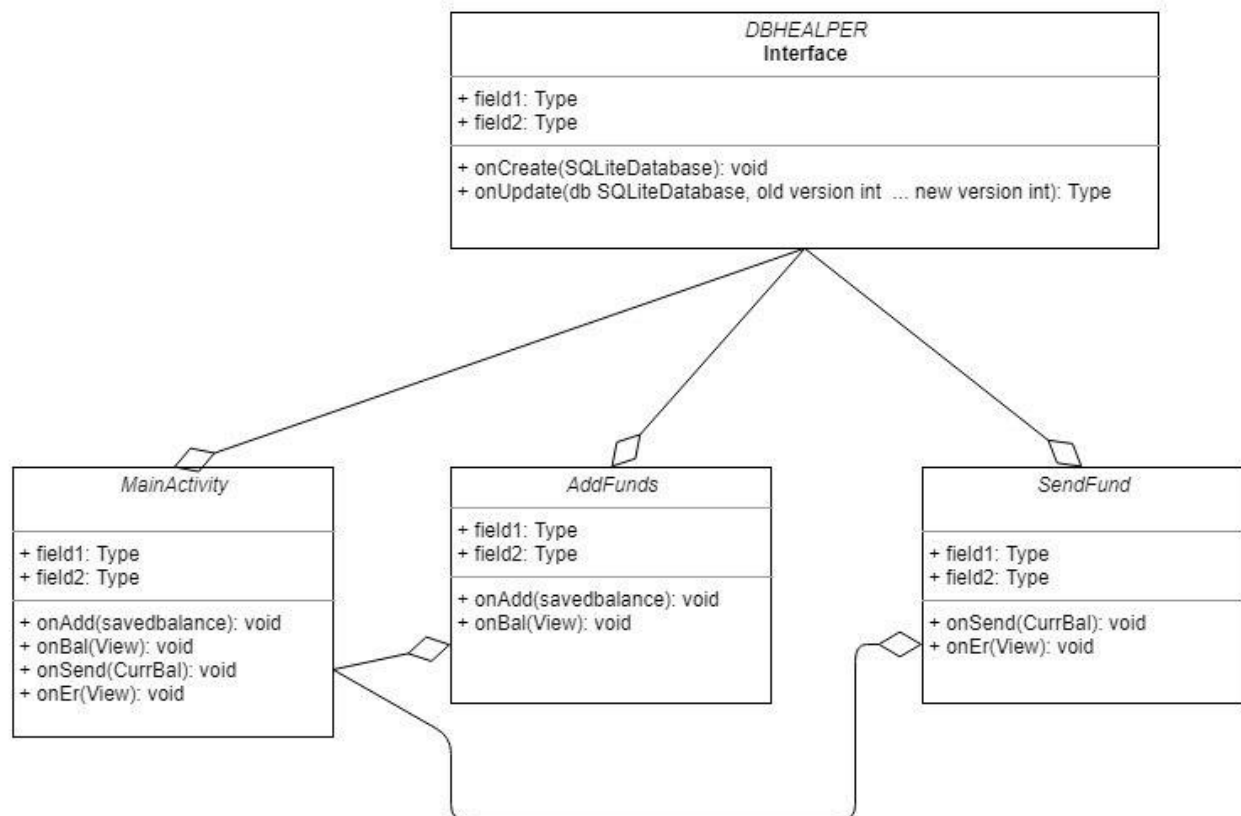
45. Vale E. HTML5. Разработка приложений для мобильных устройств. – Спб.: Питер 2015. – 225 с.

46. Xcode [Электронный ресурс] URL: <https://developer.apple.com/xcode/> (дата обращения 20.04.2020)

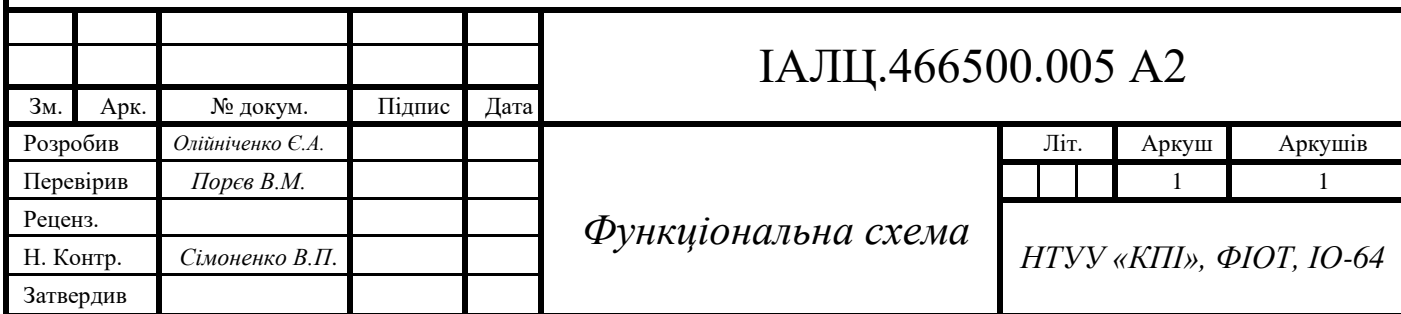
47. Биткоин / Доллар США Mode of acces: <https://ru.tradingview.com/symbols/BTCUSD/>

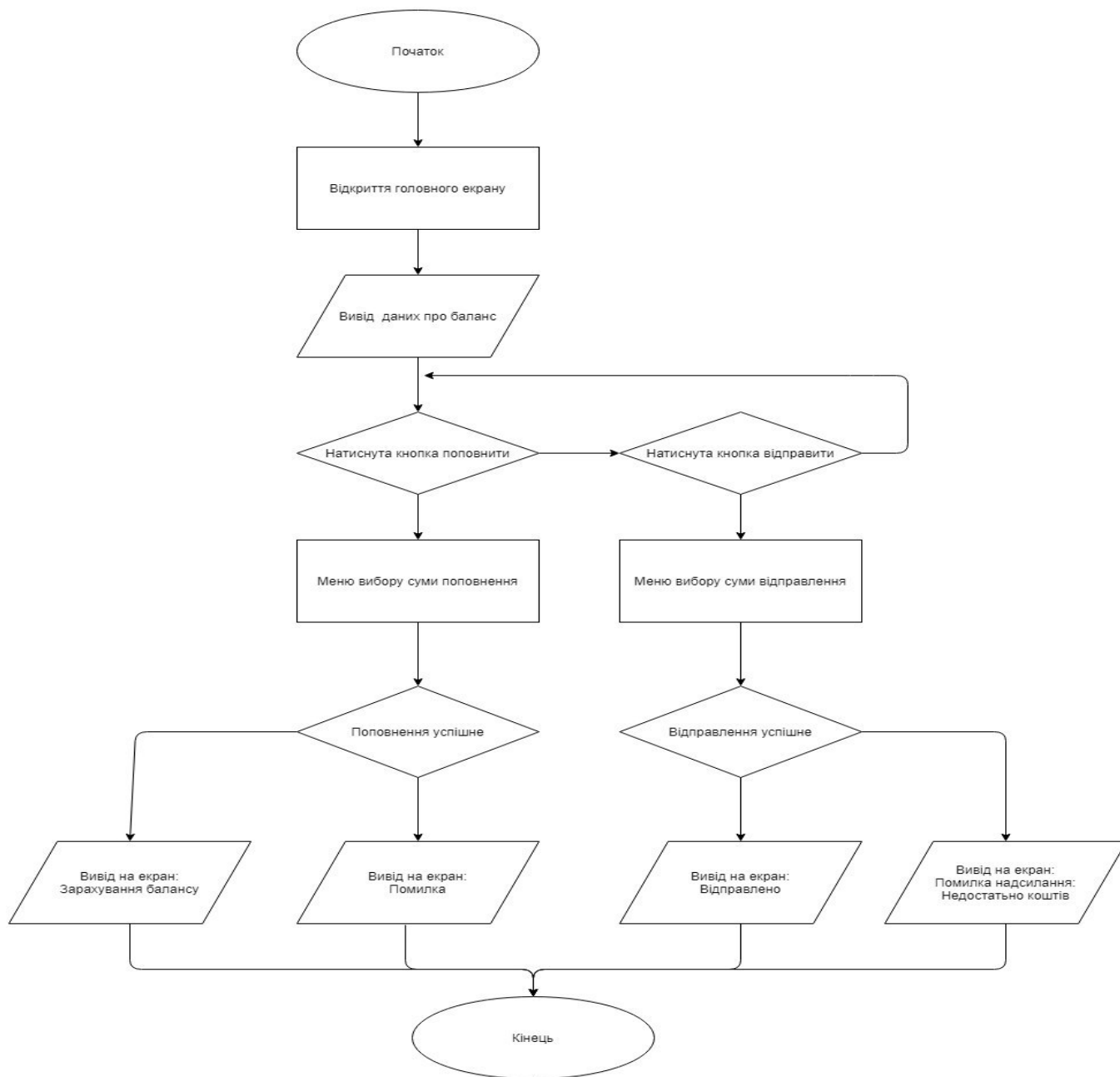
48. Рыночная капитализация Mode of acces: <https://www.blockchain.com/charts/market-cap>

					ІАЛЦ.466500.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		61



					ІАЛЦ.466500.004 А1			
Зм.	Арк.	№ докум.	Підпис	Дата	Принципова схема алгоритму			
Розробив	Олійніченко Є.А.							
Перевірів	Порєв В.М.							
Реценз.								
Н. Контр.	Сімоненко В.П.							
Затвердив					ЛІТ. Аркуш Аркушів 1 1 НТУУ «КПІ», ФІОТ, ІО-64			





					ІАЛЦ.466500.006 АЗ							
Зм.	Арк.	№ докум.	Підпис	Дата	Структурна схема				Літ.	Аркуш	Аркушів	
Розробив	Олійніченко Є.А.										1	1
Перевірив	Порєв В.М.											
Реценз.												
Н. Контр.	Сімоненко В.П.											
Затвердив												
					НТУУ «КПІ», ФІОТ, ІО-64							

Лістинг програмного коду

```
<?xml version="1.0" encoding="UTF-8"?>
<document type="com.apple.InterfaceBuilder3.CocoaTouch.Storyboard.XIB"
version="3.0" toolsVersion="14460.31" targetRuntime="iOS.CocoaTouch"
propertyAccessControl="none" useAutolayout="YES" useTraitCollections="YES"
useSafeAreas="YES" colorMatched="YES" initialViewController="HDG-Bg-x5e">
  <device id="retina5_9" orientation="portrait">
    <adaptation id="fullscreen"/>
  </device>
  <dependencies>
    <deployment identifier="iOS"/>
    <plugin identifier="com.apple.InterfaceBuilder.IBCocoaTouchPlugin"
version="14460.20"/>
    <capability name="Safe area layout guides" minToolsVersion="9.0"/>
    <capability name="documents saved in the Xcode 8 format"
minToolsVersion="8.0"/>
  </dependencies>
  <scenes>
    <!--Navigation Controller-->
    <scene sceneID="bOA-1J-vG4">
      <objects>
        <navigationController id="HDG-Bg-x5e"
sceneMemberID="viewController">
          <navigationBar key="navigationBar" contentMode="scaleToFill"
insetsLayoutMarginsFromSafeArea="NO" id="kBF-X9-Spo">
            <rect key="frame" x="0.0" y="44" width="375" height="44"/>
            <autoresizingMask key="autoresizingMask"/>
            <color key="barTintColor" white="1" alpha="1" colorSpace="custom"
customColorSpace="genericGamma22GrayColorSpace"/>
            <textAttributes key="titleTextAttributes">
              <color key="textColor" red="0.19364434480000001"
green="0.60069322589999996" blue="0.80058133600000003" alpha="1"
colorSpace="custom" customColorSpace="sRGB"/>
            </textAttributes>
          </navigationBar>
          <connections>
            <segue destination="BYZ-38-t0r" kind="relationship"
relationship="rootViewController" id="jLd-9Q-ovg"/>
          </connections>
          <navigationController>
            <placeholder placeholderIdentifier="IBFirstResponder" id="nxK-an-yOJ"
userLabel="First Responder" sceneMemberID="firstResponder"/>
          </objects>
          <point key="canvasLocation" x="-617" y="134"/>
        </scene>
    <!--View Controller-->
    <scene sceneID="tne-QT-ifu">
      <objects>
```

```

        <viewController      id="BYZ-38-t0r"      customClass="ViewController"
customModule="CryptoBTC"      customModuleProvider="target"
sceneMemberID="viewController">
        <view key="view" contentMode="scaleToFill" id="8bC-Xf-vdC">
            <rect key="frame" x="0.0" y="0.0" width="375" height="812"/>
            <autoresizingMask      key="autoresizingMask"      widthSizable="YES"
heightSizable="YES"/>
            <subviews>
                <label      opaque="NO"      userInteractionEnabled="NO"
contentMode="left"      horizontalHuggingPriority="251"      verticalHuggingPriority="251"
fixedFrame="YES" text="1000 BTC" textAlignment="center" lineBreakMode="tailTruncation"
baselineAdjustment="alignBaselines"      adjustsFontSizeToFit="NO"
translatesAutoresizingMaskIntoConstraints="NO" id="AfN-WL-G3x">
                    <rect key="frame" x="106" y="73" width="164" height="81"/>
                    <autoresizingMask key="autoresizingMask" flexibleMaxX="YES"
flexibleMaxY="YES"/>
                    <fontDescription      key="fontDescription"      type="system"
pointSize="35"/>
                    <color      key="textColor"      red="0.19364434480000001"
green="0.60069322589999996"      blue="0.80058133600000003"      alpha="1"
colorSpace="custom" customColorSpace="sRGB"/>
                    <nil key="highlightedColor"/>
                </label>
                <label      opaque="NO"      userInteractionEnabled="NO"
contentMode="left"      horizontalHuggingPriority="251"      verticalHuggingPriority="251"
fixedFrame="YES"      text="Доступно"      textAlignment="natural"
lineBreakMode="tailTruncation"      baselineAdjustment="alignBaselines"
adjustsFontSizeToFit="NO"      translatesAutoresizingMaskIntoConstraints="NO" id="LYh-Id-
WvR">
                    <rect key="frame" x="148" y="50" width="78" height="21"/>
                    <autoresizingMask key="autoresizingMask" flexibleMaxX="YES"
flexibleMaxY="YES"/>
                    <fontDescription      key="fontDescription"      type="system"
pointSize="17"/>
                    <color key="textColor" red="0.5" green="0.5" blue="0.5"
alpha="0.5" colorSpace="custom" customColorSpace="sRGB"/>
                    <nil key="highlightedColor"/>
                </label>
                <label      opaque="NO"      userInteractionEnabled="NO"
contentMode="left"      horizontalHuggingPriority="251"      verticalHuggingPriority="251"
fixedFrame="YES"      text="Bau      кошелек"      textAlignment="natural"
lineBreakMode="tailTruncation"      baselineAdjustment="alignBaselines"
adjustsFontSizeToFit="NO"      translatesAutoresizingMaskIntoConstraints="NO" id="qjE-tO-
FVN">
                    <rect key="frame" x="135" y="170" width="107" height="21"/>
                    <autoresizingMask key="autoresizingMask" flexibleMaxX="YES"
flexibleMaxY="YES"/>
                    <fontDescription      key="fontDescription"      type="system"
pointSize="17"/>
                    <color key="textColor" red="0.5" green="0.5" blue="0.5"
alpha="0.5" colorSpace="custom" customColorSpace="sRGB"/>
                    <nil key="highlightedColor"/>
            </subviews>
        </view>
    </viewController>

```

```

</label>
<label opaque="NO" userInteractionEnabled="NO"
contentMode="left" horizontalHuggingPriority="251" verticalHuggingPriority="251"
fixedFrame="YES"
text="KxFLPvYLRfP22MNAtdiGNZaVUknLosb8LDMPPhqgww6roDVUv7hW"
textAlignment="center" lineBreakMode="tailTruncation" numberOfLines="3"
baselineAdjustment="alignBaselines" adjustsFontSizeToFit="NO"
translatesAutoresizingMaskIntoConstraints="NO" id="7Lu-ps-lz3">
<rect key="frame" x="16" y="206" width="343" height="154"/>
<autoresizingMask key="autoresizingMask" flexibleMaxX="YES"
flexibleMaxY="YES"/>
<fontDescription key="fontDescription" type="system"
pointSize="30"/>
<color key="textColor" red="0.19364434480000001"
green="0.60069322589999996" blue="0.80058133600000003" alpha="1"
colorSpace="custom" customColorSpace="sRGB"/>
<nil key="highlightedColor"/>
</label>
<button opaque="NO" contentMode="scaleToFill" fixedFrame="YES"
contentHorizontalAlignment="center" contentVerticalAlignment="center"
buttonType="roundedRect" lineBreakMode="middleTruncation"
translatesAutoresizingMaskIntoConstraints="NO" id="GPY-ib-a53">
<rect key="frame" x="17" y="570" width="343" height="94"/>
<autoresizingMask key="autoresizingMask" flexibleMaxX="YES"
flexibleMaxY="YES"/>
<color key="backgroundColor" red="0.19364434480000001"
green="0.60069322589999996" blue="0.80058133600000003" alpha="1"
colorSpace="custom" customColorSpace="sRGB"/>
<fontDescription key="fontDescription" type="system"
weight="medium" pointSize="25"/>
<state key="normal" title="Пополнить на 1000">
<color key="titleColor" red="0.94509803920000002"
green="0.96078431369999995" blue="0.97254901959999995" alpha="1"
colorSpace="custom" customColorSpace="sRGB"/>
</state>
<connections>
<action selector="rechargeBtc:" destination="BYZ-38-t0r"
eventType="touchUpInside" id="EvY-kU-3Ud"/>
</connections>
</button>
<button opaque="NO" contentMode="scaleToFill" fixedFrame="YES"
contentHorizontalAlignment="center" contentVerticalAlignment="center"
buttonType="roundedRect" lineBreakMode="middleTruncation"
translatesAutoresizingMaskIntoConstraints="NO" id="JOI-rS-WNs">
<rect key="frame" x="17" y="468" width="343" height="94"/>
<autoresizingMask key="autoresizingMask" flexibleMaxX="YES"
flexibleMaxY="YES"/>
<color key="backgroundColor" red="0.19364434480000001"
green="0.60069322589999996" blue="0.80058133600000003" alpha="1"
colorSpace="custom" customColorSpace="sRGB"/>
<fontDescription key="fontDescription" type="system"
weight="medium" pointSize="25"/>

```

```

        <state key="normal" title="Омnpаæумь">
            <color key="titleColor" red="0.94509803920000002"
green="0.96078431369999995" blue="0.97254901959999995" alpha="1"
colorSpace="custom" customColorSpace="sRGB"/>
        </state>
        <connections>
            <action selector="sendBtc:" destination="BYZ-38-t0r"
eventType="touchUpInside" id="AuD-e9-0r5"/>
            <segue destination="8l3-Wl-Ra7" kind="show" id="I9d-Ro-fVf"/>
        </connections>
    </button>
</subviews>
    <color key="backgroundColor" red="1" green="1" blue="1" alpha="1"
colorSpace="custom" customColorSpace="sRGB"/>
    <viewLayoutGuide key="safeArea" id="6Tk-OE-BBY"/>
</view>
<navigationItem key="navigationItem" id="SkB-kB-CCr"/>
<connections>
    <outlet property="moneyLabel" destination="AfN-WL-G3x" id="iF4-WY-
Xg9"/>
    <outlet property="rechargeBtn" destination="GPY-ib-a53" id="Fnu-Lu-
wo0"/>
    <outlet property="sendBtn" destination="JOI-rS-WNs" id="3XL-in-
02r"/>
</connections>
</viewController>
<placeholder placeholderIdentifier="IBFirstResponder" id="dkx-z0-nzr"
sceneMemberID="firstResponder"/>
</objects>
<point key="canvasLocation" x="136.80000000000001"
y="133.74384236453201"/>
</scene>
<!--Send Controller-->
<scene sceneID="X40-7g-ZyF">
    <objects>
        <viewController id="8l3-Wl-Ra7" customClass="SendController"
customModule="CryptoBTC" customModuleProvider="target"
sceneMemberID="viewController">
            <view key="view" contentMode="scaleToFill" id="Glp-3D-kvF">
                <rect key="frame" x="0.0" y="0.0" width="375" height="812"/>
                <autoresizingMask key="autoresizingMask" widthSizable="YES"
heightSizable="YES"/>
                <subviews>
                    <label opaque="NO" userInteractionEnabled="NO"
contentMode="left" horizontalHuggingPriority="251" verticalHuggingPriority="251"
fixedFrame="YES" text="Омnpавляем на кошелек" textAlignment="natural"
lineBreakMode="tailTruncation" baselineAdjustment="alignBaselines"
adjustsFontSizeToFit="NO" translatesAutoresizingMaskIntoConstraints="NO" id="Led-15-
LCC">
                        <rect key="frame" x="89" y="120" width="196" height="21"/>
                        <autoresizingMask key="autoresizingMask" flexibleMaxX="YES"
flexibleMaxY="YES"/>

```

```

        <fontDescription          key="fontDescription"          type="system"
pointSize="17"/>
        <color    key="textColor"    red="0.5"    green="0.5"    blue="0.5"
alpha="0.5" colorSpace="custom" customColorSpace="sRGB"/>
        <nil key="highlightedColor"/>
    </label>
    <label          opaque="NO"          userInteractionEnabled="NO"
contentMode="left"    horizontalHuggingPriority="251"    verticalHuggingPriority="251"
fixedFrame="YES"    text="1AUPKzec4rNFvQ4qGqtgzHts6hwwTSs3Zt"    textAlignment="center"
lineBreakMode="tailTruncation"    numberOfLines="3"    baselineAdjustment="alignBaselines"
adjustsFontSizeToFit="NO"    translatesAutoresizingMaskIntoConstraints="NO"    id="KPj-S6-
aJy">
        <rect key="frame" x="16" y="166" width="343" height="105"/>
        <autoresizingMask    key="autoresizingMask"    flexibleMaxX="YES"
flexibleMaxY="YES"/>
        <fontDescription          key="fontDescription"          type="system"
pointSize="30"/>
        <color          key="textColor"          red="0.19364434480000001"
green="0.600693225899999996"          blue="0.80058133600000003"          alpha="1"
colorSpace="custom" customColorSpace="sRGB"/>
        <nil key="highlightedColor"/>
    </label>
    <label          opaque="NO"          userInteractionEnabled="NO"
contentMode="left"    horizontalHuggingPriority="251"    verticalHuggingPriority="251"
fixedFrame="YES"    text="Выбрать    сумму"    textAlignment="natural"
lineBreakMode="tailTruncation"    baselineAdjustment="alignBaselines"
adjustsFontSizeToFit="NO"    translatesAutoresizingMaskIntoConstraints="NO"    id="2f9-eq-
W3P">
        <rect key="frame" x="125" y="296" width="125" height="21"/>
        <autoresizingMask    key="autoresizingMask"    flexibleMaxX="YES"
flexibleMaxY="YES"/>
        <fontDescription          key="fontDescription"          type="system"
pointSize="17"/>
        <color    key="textColor"    red="0.5"    green="0.5"    blue="0.5"
alpha="0.5" colorSpace="custom" customColorSpace="sRGB"/>
        <nil key="highlightedColor"/>
    </label>
    <textField          opaque="NO"          contentMode="scaleToFill"
fixedFrame="YES"    contentHorizontalAlignment="left"    contentVerticalAlignment="center"
text="100"    textAlignment="center"    minimumFontSize="17"
translatesAutoresizingMaskIntoConstraints="NO" id="3P7-JM-q1i">
        <rect key="frame" x="95" y="329" width="185" height="155"/>
        <autoresizingMask key="autoresizingMask" flexibleMaxX="YES"
flexibleMaxY="YES"/>
        <nil key="textColor"/>
        <fontDescription          key="fontDescription"          type="system"
pointSize="25"/>
        <textInputTraits          key="textInputTraits"
keyboardType="namePhonePad"/>
    </textField>
    <button opaque="NO" contentMode="scaleToFill" fixedFrame="YES"
contentHorizontalAlignment="center"    contentVerticalAlignment="center"

```

```

buttonType="roundedRect" lineBreakMode="middleTruncation"
translatesAutoresizingMaskIntoConstraints="NO" id="D6w-zr-9jN">
    <rect key="frame" x="16" y="569" width="343" height="94"/>
    <autoresizingMask key="autoresizingMask" flexibleMaxX="YES"
flexibleMaxY="YES"/>
    <color key="backgroundColor" red="0.19364434480000001"
green="0.60069322589999996" blue="0.80058133600000003" alpha="1"
colorSpace="custom" customColorSpace="sRGB"/>
    <fontDescription key="fontDescription" type="system"
weight="medium" pointSize="25"/>
    <state key="normal" title="Омнпаѳумь">
        <color key="titleColor" red="0.94509803920000002"
green="0.96078431369999995" blue="0.97254901959999995" alpha="1"
colorSpace="custom" customColorSpace="sRGB"/>
    </state>
    <connections>
        <action selector="sendMoney:" destination="8l3-Wl-Ra7"
eventType="touchUpInside" id="5cy-P7-Hh2"/>
    </connections>
</button>
</subviews>
<color key="backgroundColor" white="1" alpha="1"
colorSpace="custom" customColorSpace="genericGamma22GrayColorSpace"/>
<viewLayoutGuide key="safeArea" id="rOj-xW-vVv"/>
</view>
<connections>
    <outlet property="sender" destination="D6w-zr-9jN" id="dnP-Lm-
QTX"/>
    <outlet property="sumToSend" destination="3P7-JM-q1i" id="1R6-GS-
d8l"/>
</connections>
</viewController>
<placeholder placeholderIdentifier="IBFirstResponder" id="2eW-qV-7n2"
userLabel="First Responder" sceneMemberID="firstResponder"/>
</objects>
<point key="canvasLocation" x="877.60000000000002"
y="133.74384236453201"/>
</scene>
</scenes>
</document>

```